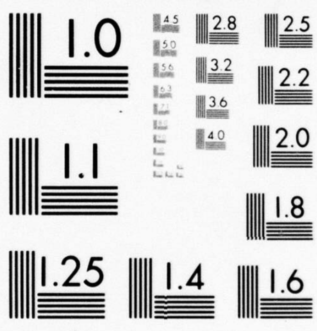
AD-A069 781 ILLINOIS UNIV AT URBANA-CHAMPAIGN COORDINATED SCIENCE LAB F/G 9/4
CODING FOR THE CONTROL OF INTERSYMBOL INTERFERENCE IN BASEBAND --ETC(U)
DEC 78 V A DIEULIIS DAAB07-72-C-0259
UNCLASSIFIED R-830 NL

1 OF 2
AD
A069781



MICROCOPY RESOLUTION TEST CHART

AD A069781

REPORT R-830 DECEMBER, 1978 UILU-ENG 78-2223

CSL COORDINATED SCIENCE LABORATORY

LEVEL

CODING FOR THE CONTROL OF INTERSYMBOL INTERFERENCE IN BASEBAND CHANNELS

VAL ANTHONY DiEULIIS

APPROVED FOR PUBLIC RELEASE. DISTRIBUTION UNLIMITED.

UNIVERSITY OF ILLINOIS – URBANA, ILLINOIS

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) (6) CODING FOR THE CONTROL OF INTERSYMBOL INTERFERENCE IN BASEBAND CHANNELS.		5. TYPE OF REPORT & PERIOD COVERED Technical Report
		6. PERFORMING ORG. REPORT NUMBER (14) R-830, UILU-ENG-78-2223
7. AUTHOR(s) (10) Val Anthony Di Euliis		8. CONTRACT OR GRANT NUMBER(s) (15) NSF ENG 75-22621 DAAB-07-72-C-0259, DAAG-29-78-C-0016
9. PERFORMING ORGANIZATION NAME AND ADDRESS Coordinated Science Laboratory University of Illinois at Urbana-Champaign Urbana, Illinois 61801		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Joint Services Electronics Program		12. REPORT DATE (11) December 1978
		13. NUMBER OF PAGES 110
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office) (9) Doctoral thesis, (12) 126p.		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Digital Communication
Baseband Channels
Intersymbol Interference
Finite-State Codes
Error-Control

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The relationship between the structure of line codes and the error probability for baseband PCM systems with intersymbol interference is studied for a broad class of finite-state-machine codes. The model for a repeatered transmission line is reduced to a digital model including the line encoder, linear dispersive digital channel, additive Gaussian noise, and a fixed threshold detector. The error probability for a first order channel model (exponential response) is then found numerically using the Gram-Charlier series representation for the Gaussian noise probability density function. In order to accomplish this computation, the

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

097 700

LB

20. ABSTRACT (continued)

moments of the intersymbol interference are derived for this channel model via a property of the encoded channel sequence statistics.

The error probability results are used to show that a simple approximation for channels which cause appreciable intersymbol interference over a span of many symbols (for example, more than fifty) provides accurate results. This approximation demonstrates the multimodal nature of the probability density of intersymbol interference, and suggests a new method of Decision Feedback Equalization which removes most of the deleterious effects of the intersymbol interference on the error probability. It is also shown that the error propagation which is inherent in Decision Feedback Equalizers can be effectively controlled with this new equalizer.

The problem of constructing codes whose objective is the reduction of intersymbol interference is also discussed. A method for comparing the error performance of different codes is proposed for the case when the decision device in the system is a fixed threshold detector. This method is based on the multimodal nature of the intersymbol interference, and is directly related to the error probability. The analysis of the error performance for different codes provides insight into the relationship between the statistics of the channel sequence generated by the encoder and the error probability at the threshold detector, and the results suggest code construction techniques which minimize the error probability. This is an improvement over the current techniques which are primarily heuristic, and consequently, do not relate the code structure directly to the error probability. An example of a code design procedure is given for this case. In addition, an example of a code design procedure is given for the Decision Feedback Equalizer mentioned previously.

Finally, the problems associated with decoding finite-state-machine block codes are discussed, and methods for detecting transmission errors at the destination terminal are considered. The existence of single error correcting finite-state-machine codes with moderate redundancy is also demonstrated.

UILU-ENG 78-2223

CODING FOR THE CONTROL OF INTERSYMBOL INTERFERENCE IN BASEBAND CHANNELS

by

Val Anthony DiEuliis

This work was supported in part by the National Science Foundation under Grant NSF ENG 75-22621 and in part by the Joint Services Electronics Program (U.S. Army, U.S. Navy and U.S. Air Force) under Contract DAAB-07-72-C-0259 and DAAG-29-78-C-0016.

CODING FOR THE CONTROL OF INTERSYMBOL INTERFERENCE IN BASEBAND CHANNELS

Val Anthony DiEuliis, Ph.D.
Coordinated Science Laboratory and
Department of Electrical Engineering
University of Illinois at Urbana-Champaign, 1978

Abstract

The relationship between the structure of line codes and the error probability for baseband PCM systems with intersymbol interference is studied for a broad class of finite-state-machine codes. The model for a repeatered transmission line is reduced to a digital model including the line encoder, linear dispersive digital channel, additive Gaussian noise, and a fixed threshold detector. The error probability for a first order channel model (exponential response) is then found numerically using the Gram-Charlier series representation for the Gaussian noise probability density function. In order to accomplish this computation, the moments of the intersymbol interference are derived for this channel model via a property of the encoded channel sequence statistics.

The error probability results are used to show that a simple approximation for channels which cause appreciable intersymbol interference over a span of many symbols (for example, more than fifty) provides accurate results. This approximation demonstrates the multimodal nature of the probability density of intersymbol interference, and suggests a new method of Decision Feedback Equalization which removes most of the deleterious effects of the intersymbol interference on the error probability. It is also shown that the error propagation which is inherent in Decision Feedback Equalizers can be effectively controlled with this new equalizer.

The problem of constructing codes whose objective is the reduction of intersymbol interference is also discussed. A method for comparing the error performance of different codes is proposed for the case when the decision device in the system is a fixed threshold detector. This method is based on the multimodal nature of the intersymbol interference, and is directly related to the error probability. The analysis of the error performance for different codes provides insight into the relationship between the statistics of the channel sequence generated by the encoder and the error probability at the threshold detector, and the results suggest code construction techniques which minimize the error probability. This is an improvement over the current techniques which are primarily heuristic, and consequently, do not relate the code structure directly to the error probability. An example of a code design procedure is given for this case. In addition, an example of a code design procedure is given for the Decision Feedback Equalizer mentioned previously.

Finally, the problems associated with decoding finite-state-machine block codes are discussed, and methods for detecting transmission errors at the destination terminal are considered. The existence of single error correcting finite-state-machine codes with moderate redundancy is also demonstrated.

CODING FOR THE CONTROL OF INTERSYMBOL INTERFERENCE IN BASEBAND CHANNELS

BY

VAL ANTHONY DIEULIIS

B.S., University of Notre Dame, 1972
M.S., University of Illinois, 1976

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Electrical Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 1978

Thesis Adviser: Professor F. P. Preparata

Urbana, Illinois

Acknowledgment

The author would like to express his appreciation to his advisor, Prof. F. P. Preparata, for introducing him to the problems of data transmission, and providing support and ideas throughout his career as a graduate student. The author is also indebted to the members of his committee, Prof. A. H. Haddad, Prof. M. B. Pursley, and Prof. D. V. Sarwate, all of whom provided advice from time to time throughout this work, and read the final draft of this thesis. The author also had the pleasure of discussing various problems with Prof. R. J. McEliece and Prof. H. V. Poor.

The authors' friend, Bita Lanys of the Slavic Review, is gratefully acknowledged for reading an early draft of the thesis, and providing very helpful editorial advice. Phyllis Youngs' emergency assistance with the details of the final draft is also greatly appreciated.

Finally, and most importantly, the author would like to thank his wife, Michele, for patiently and lovingly enduring the neglect to which she has been subject for the past months. The author intends to enjoy making up for the lost time.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

1.1 Preview of Results

In practical pulse-code modulation (PCM) data transmission systems, some form of coding is used before the data are transmitted through the channel. These codes, which are called line codes, provide redundancy for spectrum shaping, error control, and symbol synchronization in the system. The primary characteristic of the random sequences which are generated by line encoders is the correlation between the channel symbols. This correlation has a great effect on the distortion, called intersymbol interference, which is a result of the bandlimited nature of the channel. In this thesis, we study the relationship between the line code structure and the intersymbol interference. Moreover, we study the effects of this intersymbol interference on the probability of error when the channel is also disturbed by additive Gaussian noise.

The results of this study differ from those currently available in that we focus directly on the statistics of the intersymbol interference which result from the correlated sequence generated by the encoder being passed through the linear dispersive channel. In particular, we demonstrate a method for calculating the error probability for a channel with an exponential response. Our approach does not follow the current practice of truncating the impulse response of the channel to a finite number of symbols in order to calculate the probability of error. The relaxation of this assumption allows us to concentrate on the intersymbol interference statistics for channels which exhibit an appreciable response over a span of many symbols. This leads to a

simple model for channels with a slow response. Furthermore, we will see that the probability density of the intersymbol interference in this situation consists of a finite number of impulses. We will find that the multimodal nature of the intersymbol interference is related to a parameter of the channel sequence, known as the digital sum. This result is used to demonstrate a decision feedback equalizer which effectively removes the intersymbol interference. It is shown that the error propagation which is inherent to decision feedback can be effectively controlled in this new equalizer.

The insight gained from this channel model also provides a method of comparing the error performance of different line codes. This is an improvement over current techniques which rely on heuristic parameters or simulation. In addition, we are able to construct codes based directly on the minimization of the error probability. This is more desirable than the current practice of using heuristic ideas for the construction and then checking the results with simulation or calculations.

The structure of the line codes we consider also allows for the detection of transmission errors at the receiving terminal. A discussion of techniques to accomplish this is provided in this thesis. Furthermore, we demonstrate the existence of ternary line codes which are capable of correcting single errors while requiring only moderate redundancy.

1.2 Pertinent Background and Literature

PCM signals contain frequency components from dc to an upper limit which depends primarily on the rate of transmission, and therefore, PCM systems are called baseband systems. All physical channels are bandlimited, and consequently they distort any wideband signal which is transmitted through them.

When digital PCM signals are transmitted through a bandlimited medium, the distortion at every instant of time is not of fundamental importance because the digital data is recovered by sampling the received signal at regular intervals. The distortion at each sampling time, therefore, is the primary consideration. Nyquist's classic result [1] concerning this distortion provided the link between the speed with which digital data may be transmitted and the bandwidth of the medium; namely, when the symbols are transmitted at the rate of 1/T symbols/sec, the minimum bandwidth of the medium for distortionless reception is 1/2T hz. Furthermore, the frequency characteristic of this minimum bandwidth medium must be rectangular, which is physically unrealizable.

Research on data transmission systems has produced many theoretical results pertaining to methods for designing media which approximate the behavior of the distortionless baseband channel. These normally require a greater bandwidth than the theoretical minimum because of the unrealizability problem, and because of the methods used for sampling the analog signal (symbol synchronization). A general introduction to these techniques and results is provided in [2] and [3], and a collection of important papers in this area has been edited by Franks[4]. An extensive bibliography and survey may be found in [5].

A fundamental consideration in any communication system is interference from noise which is external to and independent of the information signal. The standard technique of analysis is the modeling of this noise as an uncorrelated Gaussian random process. The most natural and widely accepted criterion for the performance is the probability that a digital symbol will be received incorrectly (the probability of error). For a distortionless channel, the error probability is a function of the signal-to-noise ratio (SNR) only. However, when there is distortion, the error probability also depends on the statistics

of the distortion. These statistics in turn depend on the impulse response of the channel and the statistics of the digital signal. The most common assumptions made about these two factors are that the impulse response of the channel is nonzero for a finite length of time only, and the data symbols which are transmitted over the channel are statistically independent binary random variables. The latter assumption restricts the control of the intersymbol interference to the design of the medium (which includes pulse shaping, equalization, etc.). The results of these analyses provide various optimal pulse shaping, equalization and detection schemes. These results, however, are optimal with respect to criteria other than the error probability (mean-squared error, peak distortion) because the direct analysis of error probability has not yielded insight on how the intersymbol interference statistics affect the error probability.

The assumption of statistically independent binary random variables has another weakness. In the design of actual systems, some form of coding is used before the digital signal is transmitted. That is, a sequence of statistically independent binary random variables is transformed into another sequence of random variables, which is usually correlated. Moreover, the channel sequence is often multilevel (in particualar, ternary) because less reduction in the symbol transmission rate is necessary (or an increase in rate is possible), and the necessary increase in the signal power to overcome the reduced noise margin is not a problem. Techniques have been developed to calculate the error probability in these situations; however, the nature of the intersymbol interference is not apparent, and the relationship between the coding method and the error probability is not clear.

Two survey papers on coding techniques for baseband systems have been authored by Kobayashi[6] and Croisier[7]. These cover the majority of codes which have found practical use, including the correlative-level encoding method (also known as partial response), and sequence-state methods.

1.3 Outline of the Thesis

In Chapter 2, the appropriate models for the PCM transmission line with regenerative repeaters are presented and the problem of determining the statistics of the intersymbol interference is discussed. We also introduce a finite-state-machine model for a general class of codes which includes all known codes of practical interest. A method for determining the necessary statistics for the intersymbol interference for this class of codes is shown, and some assumptions about these statistics are discussed. An important parameter of the encoded sequence, the digital sum, is also introduced.

In Chapter 3, we derive the expressions for the error probability and apply them to a first order model of intersymbol interference. This model is a departure from the usual assumption of a finite duration channel response. In fact, the focus of our development is the nature of the intersymbol interference statistics when the effective channel response is longer than, say, 50 symbols. We illustrate the error probability results for several known codes under various channel conditions. A study of the limiting case of the first order channel as the response duration (or memory) approaches infinity is presented in Chapter 4. We will find that the intersymbol interference possesses a multimodal probability density for channels with a slow response. We show that assuming the probability density of the intersymbol interference as a sum of translated Gaussian densities provides very accurate error probability results.

Some simulation results we present also provide evidence for this assumption. Finally, a decision feedback equalization scheme which removes the multimodal nature of the intersymbol interference is discussed.

We consider the problem of synthesizing codes for the control of intersymbol interference in Chapter 5. After a preliminary discussion on the general constraints for the synthesis of this class of codes, we develop some ideas for designing codes to minimize the error probability in the presence of the multimodal intersymbol interference, and show an example of a construction. We also show an example of code construction for a system with the decision feedback equalization scheme introduced in Chapter 4.

We conclude this thesis with some considerations of the decoding procedure for these codes in Chapter 6. We also discuss error detection schemes for the decoder. Finally, a known construction of ternary error correcting codes is used to demonstrate the existence of finite state machine codes which correct single errors but require only moderate redundancy.

CHAPTER 2

BASEBAND DATA TRANSMISSION SYSTEMS

2.1 System Model

Since we are interested in analyzing the performance of encoding techniques for baseband systems, it is worthwhile to begin with a discussion of the overall system model. Fig. 1 illustrates the gross model for a baseband system with regenerative repeaters.

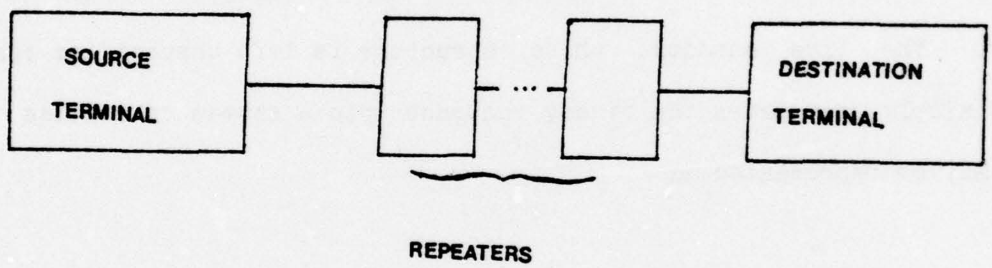


Figure 1. Overall Model of Repeatered Line

The source terminal emits the information signal in a form which is suitable for the transmission medium which is typically a twisted-pair or coaxial cable. Fig. 2 illustrates an appropriate model.

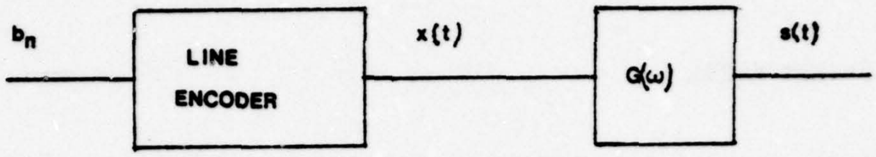


Figure 2. Source Terminal

In this analysis, we shall assume that the information sequence, b_n, is binary. The line encoder, whose structure is left unspecified for the time being, simply translates the binary sequence into a stream of impulse functions which may be represented as

$$x(t)= \sum_{k=-\infty}^{\infty} a_k \delta (t-kT), \tag{2.1}$$

where a_k is an L-ary symbol[1], and 1/T is the data rate. The sequence, $\{a_k\}$, shall be called the <u>channel</u> <u>sequence</u>. The signal, x(t), then passes through the linear system, $G(\omega)$ to produce the modulated signal

$$s(t)= \sum_{k=-\infty}^{\infty} a_k g(t-kT), \tag{2.2}$$

which is a repetitive sequence of translated causal functions, g(t), having the

---

[1]The elements of the L-ary alphabet, Λ, shall be considered as numbers such that $\Lambda =\{ \ell : |\ell| \leq L -2\}$ for L odd, and $\Lambda =\{\pm(2\ell -1)/2 : \ell=1,...,L/2\}$ for L even.

multiplier a_k for $t \geq kT$. The modulated signal, s(t), is transmitted through the cable to the regenerative repeater. We shall model the transmission medium as a linear system having the frequency response, $C(\omega)$, and correspondingly its time response, c(t), which is the inverse Fourier transform of $C(\omega)$.

A repeater model for this system is shown in Fig. 3. In this illustration $L(\omega)$ is a linear system (including amplification) which filters the incoming signal, and the timing circuit extracts information from the received signal so that the sampler is synchronized to the symbol stream. The decision element, which for cost reasons is usually a threshold slicer, produces an estimate, $\hat{a}_k$, of the symbol a_k at every time k. The estimated sequence, $\hat{a}_k$, is then passed through the pulse shaping filter, $G(\omega)$, which is identical to that in Fig. 2. The signal s(t) is now transmitted through the next section of cable until it is processed by the next repeater.

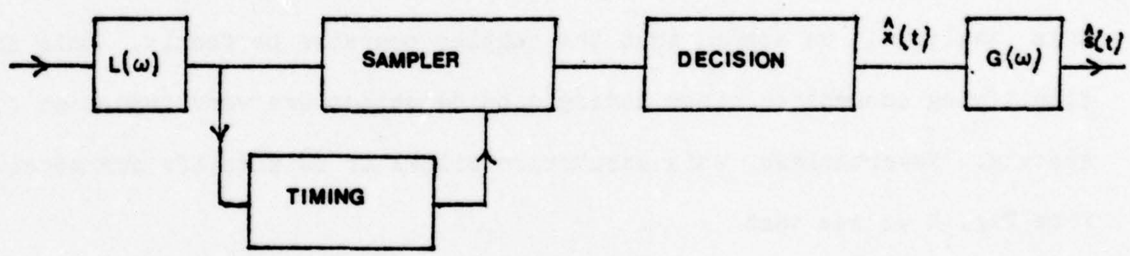


Figure 3. Repeater Model

For our purposes, it is convenient to consider the cascade of $G(\omega)$, $C(\omega)$, and the repeater front-end, $L(\omega)$, as a single element in the model. The motivation for this lies in our interest to study the relationship between the line encoder and the estimated symbol sequence, $\{\hat{a}_k\}$.

The input signal to the sampler is also corrupted by noise, $\xi(t)$, which we shall model as White Gaussian with zero mean and variance equal to $N_0/2$. These considerations allow us to model the first link in the total system by the block diagram in Fig. 4.

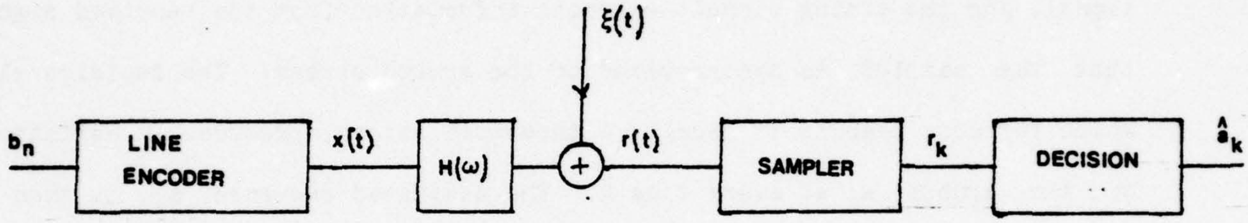


Figure 4. Simplified model of the first link

Note that we have omitted the timing circuit from the simplified model. For this analysis, we assume that the sampler operates perfectly. This is a major simplifying assumption since timing considerations are very important for these systems. Nevertheless, this assumption allows us to simplify our model further. From Fig. 4 we see that

$$r(t)=\int_{-\infty}^{t} x(\tau)h(t-\tau)d\tau + \xi(t)= \sum_{k=-\infty}^{\infty} a_k h(t-kT) + \xi(t). \tag{2.3}$$

Thus, after the sampling operation we have

$$r_k=r(kT+t_0)= \sum_{j=-\infty}^{\infty} a_j h[(k-j)T+t_0]+ \xi(kT+t_0) \tag{2.4}$$

where t_0 is a parameter in the timing circuit. Consequently, we can consider the system in Fig. 4 equivalent to a digital system. This simplification leads to the system as shown in Fig. 5,

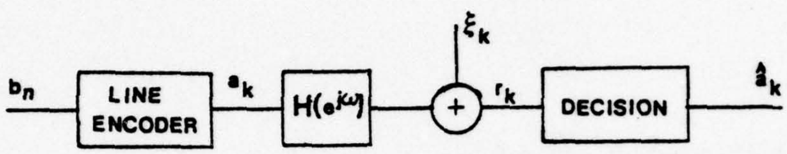


Figure 5. Equivalent digital model

where the digital system, $H(e^{j\omega})$, is derived from the cascade of $G(\omega)$, $C(\omega)$, and $L(\omega)$ by the familiar "folding" operation.

Whenever $\hat{a}_k \neq a_k$, we say that a <u>symbol error</u> has occurred; furthermore, the probability of occurrence of symbol errors is the <u>probability of error</u>, P(E). We shall consider P(E) to be the most important indicator of system performance; hence the objective of our analysis is to determine how to minimize P(E).

Errors in PAM/PCM systems are attributable to various causes. The linear channel, $H(e^{j\omega})$, produces a time dispersed sequence whereby the symbol a_k at a given time affects the received symbol sequence, r_j, for $j>k$. This is intersymbol interference (I.I.). In addition, the Gaussian noise sequence, ξ_k, further corrupts the dispersed signal. The effect of the combination of these elements (or either one singly) may be sufficient to override the operation of the decision element, thereby causing an error. These two causes of error constitute the topic of this thesis. Other causes of error which are not considered here are timing errors, cross-talk, and non-linearity in the actual

system.

2.2 Error Channel

The equivalent digital system allows us to express the received symbol at time k as

$$r_k=h_0a_k+\sum_{j\neq k} h_{k-j}a_j+\xi_k. \tag{2.5}$$

We may consider $h_0=1$ without loss of generality so that $r_k=a_k+i_k+\xi_k$ where

$$i_k=\sum_{j\neq k} h_{k-j}a_j \tag{2.6}$$

is the I.I.

For the purpose of analyzing the relationship between the line sequence and the I.I., it is convenient to substitute an equivalent model for the linear system, $H(e^{j\omega})$, in Fig. 5. At every time k, the desired signal at the detector is a_k, and the interfering signal is the response of $H(e^{j\omega})$ to the previous symbols. We can represent the impulse response of $H(e^{j\omega})$ as

$$h_j=\begin{cases} 0; & j<0 \\ 1; & j=0 \\ h_j; & j>0. \end{cases} \tag{2.7}$$

We can now view $H(e^{j\omega})$ as the parallel arrangement of a feedforward connection and a linear system having the impulse response

$$\hat{h}_j = \begin{cases} 0; & j<0 \\ & \\ h_j; & j>0. \end{cases} \tag{2.8}$$

We shall refer to the system having this impulse response as the error channel.

By means of this model, the I.I. and desired signal are separated in a conceptually simple manner, as illustrated in Fig. 6.

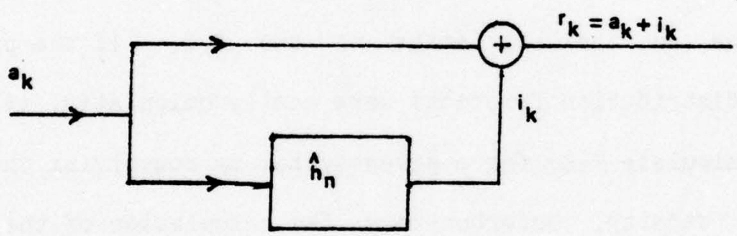


Figure 6. Error Channel

We must note that we are assuming that a channel symbol does not cause I.I. for any past symbols (i.e., the error channel is causal), although in a practical system, this may not be the case. Generally, the nonlinear phase characteristic of the channel causes the pulse to exhibit a response before the peak of the received signal. This response interferes with past symbols because the samples of the received signal are taken near the peak of the response. The non-causal response of the error channel arises because the inherent linear delay in the system is not a factor in the analysis. However, in such a system a symbol will interfere with a finite number of past symbols but may interfere with an

infinite number of future symbols. The analysis that follows will concentrate on the latter case.

2.3 Error Probability: Mathematical Difficulties

We are interested in calculating P(E) for the system model given in Fig. 5. When $\hat{h}_j=0$ for all j (i.e., there is no I.I.), the calculation is straightforward and we obtain the familiar dependence of P(E) on the signal-to-noise ratio. The conditions on $H(e^{j\omega})$ which ensure $\hat{h}_j=0$ are the well known Nyquist criteria.

Whenever the I.I. is not zero, the calculation of P(E) is complicated by its dependence on the statistics of the I.I. If the probability density function (or distribution function) were easily calculated, it would be a simple matter to calculate P(E) for a given system by convolving the Gaussian density with the I.I. density. Unfortunately, the calculation of the I.I. probability distribution is quite a difficult problem. In fact, very little can be said about the probability distribution in general. In [8], a survey/tutorial of results is presented on the problem. If the error channel has a finite response, then it is clear that the probability distribution is a staircase function, since the I.I. may take on a finite number of values only. However, when the error channel has an infinite response, the distribution may or may not be continuous, depending on the rate of decay of the response. Furthermore, even if the distribution is continuous, it may not be differentiable (so that the density may not exist). Despite closed-form results which have been found for some special cases, the problem remains open and the possibility of simple results seems remote. From the viewpoint of applications, therefore, numerical methods for finding P(E) appear to be necessary.

In the past decade, much effort has been expended on finding upper/lower bounds and approximations for P(E), and the available results provide a myriad of techniques of varying complexity and accuracy. A survey of the pre-1975 literature in this area is given in [9]. Later we shall utilize one such method which is an approximation of P(E) based on an infinite series representation of the joint I.I. and noise probability distribution (Gram-Charlier series). This method requires knowledge of the moments of the I.I. only, which may be found recursively without an analytical expression for the distribution function. Before discussing the application of this technique, we shall present a description of a broad class of line encoders which encompasses most situations of practical interest.

2.4 Finite State Machine Line Encoders

In the description of the overall system model, we did not indicate the structure of the line encoder (Fig. 2). The subsequent discussion of the P(E) will show that the statistics of the I.I. are strongly dependent on the statistics of the channel sequence. A class of encoders for which the statistics may be evaluated is described by means of a finite state machine (FSM) model. Let the quintuple $\{\mathcal{B}, \mathcal{L}, \mathcal{S}, f, g\}$ denote the encoder with $\mathcal{B}=\{0,1\}^m$, which are inputs consisting of sequences of m binary symbols (input blocks), $\mathcal{L} \subset \Lambda^n$, outputs consisting of sequences of n L-ary[(2)] symbols (output blocks or codewords), and $\mathcal{S}=\{1,2,\ldots,r\}$, a finite set of states conveniently denoted as integers. The output function, $f: \mathcal{B} \times \mathcal{S} \to \mathcal{L}$, can be described as a set

---

(2) See footnote (1).

of functions $F=\{f_1,f_2,\ldots,f_r\}$, where $f_i: \mathcal{B}\to\mathcal{L}$. The mapping f_i shall be referred to as the encoding mode corresponding to state i. The state transition function, $g: \mathcal{B}\times\mathcal{S}\to\mathcal{S}$, determines the state of the encoder at all times given that the initial state is known.

The input block sequence is assumed to be stationary with each block statistically independent. For each b_u; $u=1,\ldots,2^m=K$, its associated probability will be denoted θ_u, with the obvious condition

$$\sum_{u=1}^{K} \theta_u = 1 .$$

The encoded process (channel process) may be represented as in Fig. 7 below.

$$\cdots \mid a_0^{(1)} a_0^{(2)} \cdots a_0^{(n)} \mid a_1^{(1)} a_1^{(2} \cdots a_1^{(n)} \mid \cdots \mid a_i^{(1)} a_i^{(2)} \cdots a_i^{(n)} \mid \cdots$$

Figure 7. Channel Sequence

This representation shows the symbol sequence properly framed with respect to the words of length n. The symbol a_k represents the 1^{th} digit of the k^{th} codeword in the channel sequence. We will refer to the time of transition from word $\underline{a}_{i-1}$ to $\underline{a}_i$ (i.e., the symbol transition $a_{i-1}^{(n)}$ to $a_i^{(1)}$) as wordtime i. Note that the sequence representation explicitly shows the state sequence of the encoder. We shall also adopt the convention that the state transition from s_{i-1} to s_i occurs at wordtime i.

At wordtime i, a new binary blc . is input to the encoder and the state transition function changes the internal state of the encoder according to $s_i=g(b_{i-1},s_{i-1})$(3). Assuming that the binary input process is block stationary with statistically independent elements, it easy to show that the state sequence is a homogeneous Markov chain with stationary transition probabilities (see, e.g., [10]). Let $\mathcal{B}_{ij}$ be the set of input words which cause the encoder to change from state i to state j (i.e., $\mathcal{B}_{ij}=\{b\varepsilon\mathcal{B}\,|\,j=g(b,i)\}$), and denote the one-step transition matrix of the state process as $\Pi\overset{\Delta}{=}||\pi_{ij}||$, then

$$\pi_{ij}=\sum_{b_u\varepsilon\mathcal{B}_{ij}}\theta_u \tag{2.9}$$

A restriction on Π (hence, on the state transition function, $g(b,s)$) is that the state process must be a <u>fully regular</u> homogeneous Markov chain. For this class of state transition matrices, there exists $\Pi_\infty=\lim_{k\to\infty}\Pi^k$ where Π^k denotes the matrix of k-step transition probabilities. Furthermore, Π_∞ consists of identical rows $\underline{p}=(p_1,p_2,\ldots,p_r)$ such that $\underline{p}\Pi=\underline{p}$. The vector $\underline{p}$ is called the stationary probability distribution. This restriction does not affect the application of the model to practical situations.

A subclass of the FSM encoders which has useful properties results from considering a specific form for the state transition function, g. Let $\underline{a}=(a^{(1)},a^{(2)},\ldots,a^{(n)})$ be a codeword and define the <u>characteristic</u> of $\underline{a}$, $\gamma(\underline{a})$, as

---

(3)Here and hereafter, s_i and b_i refer to the state and input at wordtime i respectively.

$$\sum_{i=1}^{n} a^{(i)} ,$$

where the sum is real. Furthermore, we assume there is a function $\sigma: S \to Z$ (Z is the set of integers) such that for every $s \varepsilon S$, and $b \varepsilon B$, we have

$$\sigma[g(b,s)] = \sigma(s) + \gamma[f(b,s)]. \quad (2.10)$$

Expression (2.10) means that the image of the next state under σ is equal to the image of the present state plus the characteristic of the present codeword. Codes with this property have been called <u>counter-encodable</u> (CE) [11] because the encoder can be implemented with a digital counter which contains the value $\sigma(s)$ for the current state s.

A simple example of a ternary CE code is the Paired Selected Ternary (PST) [12] which is described by the codebook in Table 1.

Table 1. Codebook for PST

u	b_u	state 1	state 2
1	00	+-	+-
2	01	-+	-+
3	10	+0	-0
4	11	0+	0-

We note from Table 1 that the code maps blocks of 2 binary digits (m=2) into blocks of 2 ternary digits (n=2), and that there are two encoding modes

corresponding to the two states ($r=2$). We shall refer to an FSM code as an (m,n,r) code whereby PST is a (2,2,2) code. Note that state 1 contains two words each with characteristic equal to zero (viz. +- and -+) and two words each having characteristic equal to one (viz. +0 and 0+). Likewise in state 2 there are 2 words having characteristic equal to zero and 2 words having characteristic equal to minus-one. As an example of the encoder operation, if at wordtime n, the encoder is in state 1 and the binary block '10' is input, the encoder will output the block '+0' and change to state 2. The function σ in this case is simply $\sigma(i)=i$; $i=1,2$. If the input blocks are assumed to be equiprobable, the state transition matrix is

$$\Pi = \begin{bmatrix} 1/2 & 1/2 \\ 1/2 & 1/2 \end{bmatrix}$$

2.5 Statistics of Intersymbol Interference for FSM Codes

The previous discussion of the encoder structure has assumed that the binary input process is block stationary. The state process, therefore, is also stationary since $s_{i+1}=g(b_i,s_i)$, and the codeword process is block stationary since $\underline{a}_i=f(b_i,s_i)$. However, we are interested in the channel symbol sequence statistics. When the channel sequence is produced by a block encoder, the stream of weighted impulses which is transmitted through the channel (see (2.1)) may be represented as

$$x(t)= \sum_{k=-\infty}^{\infty} \sum_{l=1}^{n} a^{(l)}\, \delta[t-kn-(l-1)] \qquad (2.11)$$

where the index k indicates the codeword position in the sequence, and the index l indicates the codeword digit (phase). The process (2.11) is cyclostationary with period n because the codeword sequence is stationary.

When the discrete process (2.11) is applied to the error channel, the resulting I.I. will be cyclostationary with period n in the steady state. Thus we can view the I.I. as block (or vector) stationary. This allows us to derive the statistics of the I.I. by considering the statistics of a single block. Let $\underline{i}_k=(i^{(1)}_k, i^{(2)}_k,\ldots,i^{(n)}_k)$ denote the I.I. vector which corresponds to the k^{th} codeword, $\underline{a}_k$ (i.e., each component of the vector interferes with the corresponding symbol in the codeword). The joint probability distribution for the I.I., denoted by $F_I(\underline{\alpha})=\text{Prob}(\underline{i}_k<\underline{\alpha})$, is a function of the statistics of the set of possible codeword sequences, $A=\underline{a}_k,\underline{a}_{k-1},\ldots$, which we denote by $\mathcal{A}$. When the encoder is an FSM we may conveniently partition the set $\mathcal{A}$ into r subsets. $\mathcal{A}_i$, which are defined as

$$\mathcal{A}_i=\{A \mid s_{k+1}=i\}. \tag{2.12}$$

The event $\mathcal{A}_i$ is, therefore, identical to the event $s_{k+1}=i$, and the probability distribution function of the I.I. is

$$F_I(\underline{\alpha})=\sum_{i=1}^{r} F_I(\underline{\alpha}\mid s_{k+1}=i)P(s_{k+1}=i), \tag{2.13}$$

which is well defined since the state process possesses a steady state distribution. The probability distribution may be calculated with a finite sum as a result of the encoder structure.

Of course, (2.13) does not indicate how to find the conditional probability distributions $F_I(\alpha | s=i)$. Rather than pursue this line of analysis, we shall apply this simplification to the case of CE codes in the following section.

2.6 Digital Sum Process and CE codes

Thus far we have discussed the channel sequence and the I.I. sequence as stationary vector processes. In the discussion of the system model, it was noted that the decision element in a regenerative repeater is usually a simple symbol-by-symbol threshold detector. Therefore we need to develop the statistics for the symbol process rather than the vector process. We have noted in the previous section that the joint distribution of the codeword process is specified by conditioning on the encoder state. We also know that the current I.I. vector is specified by the current state (i.e., it is statistically independent of the previous codewords if the state is known). However, we have not addressed the problem of finding the conditional distribution $F_I(\underline{i}_k | s_k)$. In this section, we shall narrow our analysis to CE codes, and propose some simplifying assumptions about the symbol sequence statistics. These assumptions will allow us to derive conditional distributions which are closely related to $F_I(\underline{i}_k | s_k)$ in a straightforward manner.

For the remainder of this analysis we shall consider only CE codes. A useful quantity in the study of CE codes is the Digital Sum (DS). For the channel (symbol) sequence, $\ldots a_{-1}, a_0, a_1, \ldots$, the DS is defined at each time k as

$$\sigma_k \triangleq \sum_{j=-\infty}^{k-1} a_j, \tag{2.14}$$

where the sum is real. Recall that in the definition of a CE code, (2.10), the change in the encoder state was

$$\sigma(s_{k+1}) - \sigma(s_k) = \sum_{i=1}^{n} a^{(i)}, \tag{2.15}$$

where σ is a function from the state set to the integers. From (2.14) and (2.15) we see that the image under σ of the state at word time k differs from the DS at that time by, at most, an additive constant; that is

$$\sigma(s_{k+1}) = \sigma_{k+1}^{(1)} + C , \tag{2.16}$$

with $\sigma_{k+1}^{(1)}$ denoting the DS associated with the 1^{st} digit of the $(k+1)^{st}$ codeword in the sequence. Clearly, the DS process for a CE code is closely related to the state process of the encoder. Letting d be the number of DS values that may occur and recalling that r is the number of encoder states, it is obvious that $d \geqslant r$, since $\sigma^{(i)}$ does not necessarily equal $\sigma(j)$ for any $j=1,\ldots,r$ when $i \geqslant 2$. We may transform any CE codebook into an equivalent DS table. As an example, we show the DS table for PST in Table 2 (see codebook in Table 1).

Table 2. DS table for PST

u	b_u	state 1	state 2
1	00	01	12
2	01	0$\underline{1}$	10
3	10	01	10
4	11	00	11

In this case we have chosen $C=1$ (recall that $\sigma(i)=i$, $i=1,2$). Note that in the 1st digit slot $\sigma^{(1)}_{k+1}=\sigma(i)-1$ for all u. We also find that even though there are two states in the encoder, the DS may take on four different values (viz. -1,0,1,2); i.e., r=2 and d=4.

In order to determine the statistics of the I.I. for a CE code, it will be useful to introduce some ideas about the statistics of the DS process. Since the channel sequence may be placed in a 1-1 correspondence with the DS process, the DS process is also cyclostationary with period n. At this point, we introduce a simplifying assumption. Let $\sigma^{(k)}_{iu}$ denote the DS value corresponding to the k^{th} digit of the codeword which is the image of the input word b_u, when the encoder is in state i. For example, in Table 2 we find that $\sigma^{(2)}_{13}=1$ for the PST code. We shall <u>define the DS symbol process</u>, $\ldots\ \sigma_{-1}\ \sigma_0\ \sigma_1\ \ldots$, to be *stationary* with the probability mass function (pmf)

$$P(\sigma=j)=(1/n)\sum_{q=1}^{n} P(\sigma^{(q)}=j) \tag{2.17}$$

where $\sigma^{(q)}$ denotes the q^{th} component of the DS vector. This pmf may be derived from the DS table by averaging over the states and inputs as the following:

$$P(\sigma=j)=(1/n)\sum_{q=1}^{n}\sum_{i=1}^{r}\sum_{u=1}^{K} p_i\,\theta_u\,\phi^{(q)}_{iu}(j) \tag{2.18}$$

where

$$\phi^{(q)}_{iu}(j) \triangleq \begin{cases} 1; & \sigma^{(q)}_{iu}=j \\ 0; & \text{otherwise} \end{cases} \tag{2.19}$$

is a counting function. The probabilities, p_i and θ_u , have been defined in section 2.4. Our definition of the stationary symbol process may be arrived at by considering the phase of the actual symbol process as a random variable which is independent of the symbol process with the probability of any phase being 1/n.

The transition probabilities, $P[\sigma_t=j|\sigma_{t-1}=k]$, will be defined in terms of the joint process, $\{\sigma_t,a_t\}$. Since

$$\sigma_t= \sigma_{t-1}+a_{t-1}, \tag{2.20}$$

it is clear that

$$P[(\sigma_t=j),(\sigma_{t-1}=k)]= P[(a_{t-1}=j-k),(\sigma_{t-1}=k)]. \tag{2.21}$$

We may combine the codebook and DS table for any CE code into a joint code symbol/DS table by inserting the pair $(\sigma_{iu}^{(q)}, \alpha_{iu}^{(q)})$ into the appropriate position of the codebook. For the PST code (Table 1) with DS table in Table 2, we obtain the joint DS/symbol table as shown below in Table 3.

Table 3. Joint symbol/DS table for PST

u	b_u	state 1	state 1
1	00	(0,+),(1,-)	(1,+),(2,-)
2	01	(0,-),(1,+)	(1,-),(0,+)
3	10	(0,+),(1,0)	(1,-),(0,0)
4	11	(0,0),(0,+)	(1,0),(1,-)

For the PST code (moreover, for any ternary code), it is clear that $P[\sigma_t = j \mid \sigma_{t-1} = i]$ is non-zero for $j \varepsilon \{i+1, i, i-1\}$ only.

The joint probabilities in (2.21) may be calculated from (2.18) by redefining the counting function as

$$\phi_{iu}^{(q)}(k,j) = \begin{cases} 1; & \sigma_{iu}^{(q)} = k \text{ and } \alpha_{iu}^{(q)} = j-k \\ 0; & \text{otherwise} \end{cases} \tag{2.22}$$

The DS transition probability is then easily calculated from the joint probabilities (2.21) and the pmf of the DS (2.18). We define the matrix of DS transition probabilities as $Q = \| q_{ij} \|$ where $q_{ij} = P[\sigma_t = j \mid \sigma_{t-1} = i]$.

We now propose a simplifying assumption concerning the I.I. symbol process. The motivation for this assumption may be found by recalling that the vector I.I. probability distribution may be specified in terms of the encoder state. For a CE code, (2.16) indicates that this is equivalent to saying

$$F(i_t^{(1)} \mid s_t = j) = F_I(i_t^{(1)} \mid \sigma_t^{(1)} = \sigma(j)). \tag{2.23}$$

In words, (2.23) means that the probability distribution for the first component of the I.I. vector is specified by the DS that corresponds to that symbol time. Of course, this is not true for the other components of $\underline{i}_t$. Because the DS will always be finite for a CE code with a finite number of states, it may be utilized as a conditional event for the I.I. symbol process in the same manner as the encoder states may be used for the vector process. The cyclostationarity of the I.I. symbol process requires the computation of the I.I. probability

distributions for each of the n phases. However, we may define a stationary random process based on the cyclostationary I.I. process by averaging the conditional probabilities (conditioned on the DS and phase) over the phases (assuming each phase is equiprobable). We expect that the statistics of this stationary process will be similar to those of each phase of the I.I. cyclostationary process (i.e., we are assuming that each phase of the I.I. conditioned on the DS have approximately the same statistics). Hence, for the remainder of this analysis, we will assume that the probability distribution of the I.I. is independent of the codesymbol process when conditioned on the DS. In Chapter 4, we will present analysis and results which yield evidence for the validity of this assumption.

Based on these assumptions we may develop the statistics of the I.I. process. Following the development in section 2.5, we let A denote an infinite sequence of channel symbols, $a_k, a_{k-1}, \ldots$, and $\mathcal{A}$ the set of all such sequences. We can partition the set $\mathcal{A}$ into d subsets $\mathcal{A}_i$ where

$$\mathcal{A}_i = \{ A \mid i = \sum_{j=-\infty}^{k-1} a_j \} = \{ A \mid \sigma_k = i \}. \tag{2.24}$$

Thus, the probability distribution function of the I.I. based on the stationary DS/channel symbol process may be expressed as

$$F_I(\alpha) = \sum_{i=1}^{d} F_I(\alpha \mid \sigma_k = i) P(\sigma_k = i). \tag{2.25}$$

Expression (2.25) indicates that for CE codes, the I.I. distribution may be found by conditioning on a finite number of events. The topic of the next chapter will be the calculation of the conditional distributions $F_I(\alpha \mid \sigma)$. One must keep in mind that when we discuss these distributions, we are referring

to the I.I. process which results from the assumptions made in this section.

2.7 System Model Revisited

In section 2.1, we began with an overall model of a repeatered line. The source terminal, and repeater models were discussed. The remainder of this chapter was concerned with the preliminaries to evaluating the error performance of the first link in Fig. 1. We have seen that the statistics of the line encoder are of paramount importance in determining the statistics of the I.I.

The remaining links in the repeater chain are identically modeled. However when evaluating the I.I. statistics for the remaining links it should be noted that we must assume that no errors have occurred in the previous links. If errors have occurred, the statistics of the line sequence in the succeeding links will be different and the results of the analysis invalidated. The error rate in these systems is typically very low, hence, this assumption should not cause much inaccuracy in the results.

The destination terminal in Fig. 1 has not been discussed. Its digital model is shown in Fig. 8.

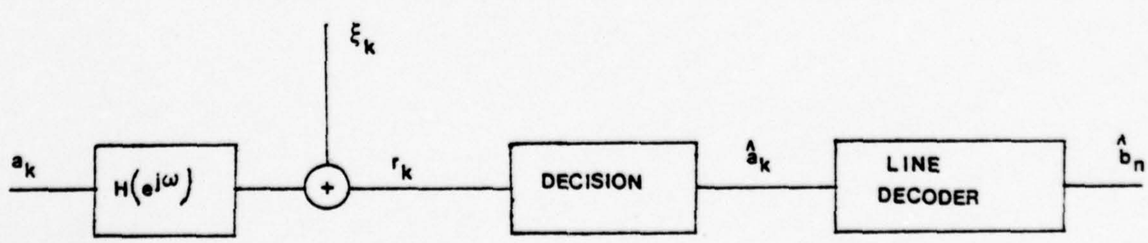


Figure 8. Model for destination terminal

It is seen to be identical to the repeater model with the exception of a line decoder which converts the estimate, a_k, to the binary estimate, b_t, of the original information sequence $\{b_t\}$. When $b_t \neq b_t$, a bit error is said to occur. The relationship between the symbol error probability and the bit error probability will be discussed later along with a discussion of the line decoder structure for FSM codes. A final note on the destination terminal concerns the decision device. Although the repeaters require simple detectors for economic and space reasons, the terminal decision device may be somewhat more complex. In addition, in order for the decoder to operate properly on block codes, the symbol stream must be frame-synchronized. We will discuss these aspects of the decoding process in a later chapter.

CHAPTER 3

FIRST ORDER MODEL OF INTERSYMBOL INTERFERENCE

3.1 Preliminary Remarks

From the discussion in section 2.3, we concluded that numerical methods are necessary to find the P(E) caused by Gaussian noise and I.I. A convenient method developed independently by Ho/Yeh [13] and Shimbo/Celebiler [14] utilizes an infinite series representation of the Gaussian density, known as the Gram-Charlier series (see, e.g., [15]). The result is an approximation to P(E) by truncating the series. The approximation may be made arbitrarily accurate by including more terms in the calculation. This method is convenient because only the moments of the I.I. are needed in the series, and these moments may be found recursively. In the above references the channel sequence was assumed to be a sequence of independent and identically distributed binary random variables. Ho and Yeh [16] extended this to include multilevel sequences. In addition, the authors have assumed that the error channel has a finite impulse response. In [17], Cariolaro and Pupolin extended the recursive method of calculating the I.I. moments to include channel sequences generated by an FSM encoder; they also assumed, however, that the error channel has a finite response. In this chapter, we shall develop a method of recursively calculating the I.I. moments for a first order channel with an exponential response (recursive channel) for CE codes. We shall take advantage of the conditioning on the DS as developed in section 2.6.

The error probability will be calculated for a fixed threshold detector. When L is odd, the threshold detector may be described by the functional relationship

$$\hat{a}_k = \begin{cases} L-2; \; r_k > (2L-5)/2 \\ i; \; (2i-1)/2 \leq r_k \leq (2i+1)/2, \; k=0,\pm1,\ldots,\pm(L-3) \\ -(L-2); \; r_k < -(2L-5)/2 \; . \end{cases} \tag{3.1}$$

We may define a similar relationship when L is even, but, for the remainder of this thesis we shall focus on the ternary case, (L=3). For the ternary case (3.1) becomes

$$\hat{a}_k = \begin{cases} 1; \; r_k > 1/2 \\ 0; \; -1/2 \leq r_k \leq 1/2 \\ -1; \; r_k < -1/2 \; . \end{cases} \tag{3.2}$$

It is clear from (3.2) that the threshold detector will make an error whenever

$$\begin{aligned} &\text{i)} \quad r_k < 1/2, \text{ and } a_k=1 \\ &\text{ii)} \quad r_k \geq 1/2 \text{ or } r_k \leq -1/2, \text{ and } a_k=0 \\ &\text{iii)} \quad r_k > -1/2, \text{ and } a_k=-1 \, . \end{aligned} \tag{3.3}$$

Recalling that $r_k = a_k + i_k + \xi_k$, we may express the error probability as

$$P(E) = \begin{cases} P(i_k + \xi_k < 1/2); \; a_k=1 \\ P(i_k + \xi_k \geq 1/2) + P(i_k + \xi_k \leq -1/2); \; a_k=0 \\ P(i_k + \xi_k > 1/2); \; a_k=-1 \, . \end{cases} \tag{3.4}$$

At this point, it is important to note that since the channel symbol sequence and the I.I. sequence are cyclostationary, the sequence of error probabilities

(the <u>error process</u>) is also cyclostationary. The actual error performance of the system can be characterized by the vector $\underline{e}=(e^{(1)},e^{(2)},\ldots,e^{(n)})$, where

$$e^{(j)}=\begin{cases} P(i^{(j)}+\xi^{(j)}<1/2);\ a^{(j)}=1 \\ P(i^{(j)}+\xi^{(j)}\geqslant 1/2)+P(i^{(j)}+\xi^{(j)}\leqslant -1/2);\ a^{(j)}=0 \\ P(i^{(j)}+\xi^{(j)}>1/2);\ a^{(j)}=-1 \end{cases} \tag{3.5}$$

is the error probability when the j^{th} digit of a codeword is being estimated. The average error probability may then be defined as the arithmetic mean of the components of $\underline{e}$:

$$P(E)=(1/n)\sum_{j=1}^{n} e^{(j)}. \tag{3.6}$$

It is clear that the computation of (3.6) requires finding the statistics of the <u>vector</u> I.I. process. However for CE block codes, we have assumed that we can determine the statistics of the I.I. by conditioning on the DS. This simplifies the analysis greatly but it is important to remember that the error probabilities that we derive will not be identical to (3.6), except when the blocklength n is equal to one.

We shall find P(E) via (3.4) with the symbol stationarity assumption. Letting $F_{I+N|a}(\varepsilon|\alpha)=P(i_k+\xi_k<\varepsilon \mid a_k=\alpha)$ where $\alpha\in\{-1,0,1\}$ be the probability distribution of the noise plus I.I. conditioned on the symbol, a_k, the expression for P(E) may be written as

$$P(E)=F_{I+N|a}(-1/2|1)P(a_k=1)+\overline{F}_{I+N|a}(1/2|-1)P(a_k=-1)+ \\ \{F_{I+N|a}(-1/2|0)+\overline{F}_{I+N|a}(1/2|0)\}P(a_k=0) \tag{3.7}$$

where $\overline{F}_{I+N|a}(\varepsilon|\alpha)=1-F_{I+N|a}(\varepsilon|\alpha)$. Because the noise is independent of the symbol sequence and the I.I., we need be concerned with the I.I. distributions, $F_{I|a}(\varepsilon|\alpha)$. This distribution may be expressed as

$$F_{I|a}(\varepsilon|\alpha)=\sum_{j=1}^{d} F_{I,\sigma|a}(\varepsilon,j|\alpha)P(a=\alpha|\sigma=j)= \sum_{j=1}^{d} F_{I,\sigma|a}(\varepsilon,j|\alpha)q_{j,j+\alpha} \tag{3.8}$$

where $F_{I,\sigma|a}(\cdot,\cdot|\cdot)$ is the joint distribution of the I.I. and the DS conditioned on the channel symbol. However we have assumed that the I.I. distribution is completely specified by the DS at anytime so that $F_{I,\sigma|a}=F_{I,\sigma}$. With this simplification, we may condition the error probability on the DS, whereby we obtain

$$P(E|\sigma=j)=F_{I+N|\sigma}(-1/2|j)[q_{j,j}+q_{j,j+1}]+ \overline{F}_{I+N|\sigma}(1/2|j)[q_{j,j}+q_{j,j-1}]. \tag{3.9}$$

The error probability expression (3.7) may then be expressed in terms of (3.9) as follows

$$P(E)=\sum_{j=1}^{d} P(E|j)P(\sigma=j) \tag{3.10}$$

Before evaluating (3.10), it is necessary to derive the conditional distributions, $F_{I|\sigma}(\varepsilon|j)$. In the next section, we shall do this for a first order channel.

3.2 I.I. for the First Order Channel

In this section we shall analyze the statistics of intersymbol interference when CE codes are transmitted through a channel containing at most one pole in its transfer function. We assume that the channel's impulse response is described as

$$h_j = \begin{cases} 0; & j<0 \\ 1; & j=0 \\ c\,\lambda^{j-1}; & j>1 \end{cases} \tag{3.11}$$

where $0<\lambda<1$. This description of the channel is useful because it allows us to separate the desired signal component from the I.I. easily. This can be seen in the block diagram model of the system shown in Fig. 9.

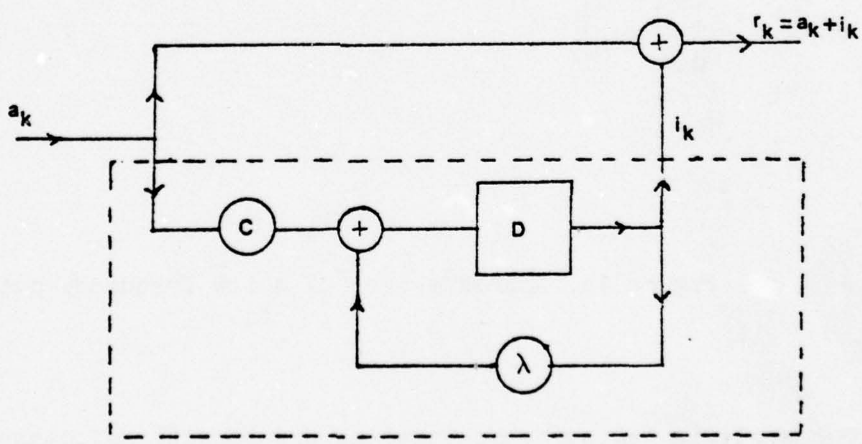


Figure 9. First order channel.

The figure clearly shows that the received signal is the sum of the desired symbol and the I.I. which is generated by the error channel (enclosed in the dotted box).

Furthermore, by varying the value of c we can approximate different channel characteristics. In particular, when c is negative, the response of the system approximates that of a channel with a low frequency cutoff. Fig. 10 illustrates this situation.

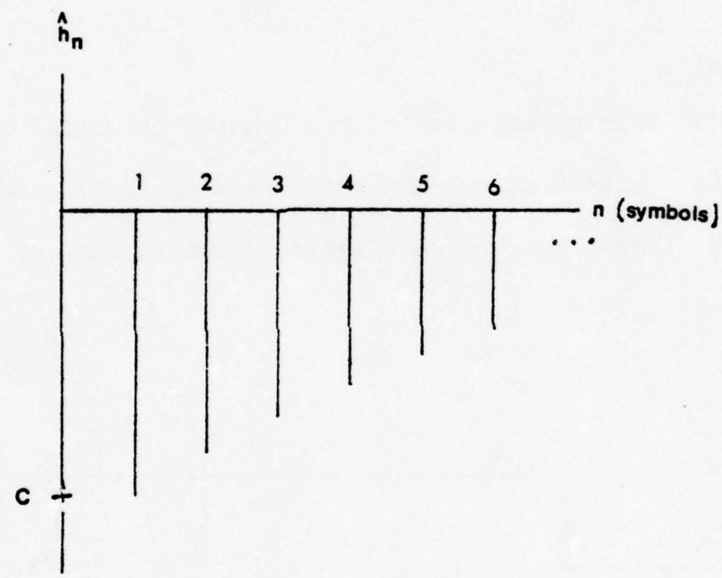


Figure 10. Approximation of a low frequency cutoff.

Our analysis begins by considering the conditional distribution of the I.I. on the DS, $F_{I|\sigma}(\varepsilon|k)$. For the derivation to follow, it will be conceptually simpler to explicitly note the time dependence on the distributions. We shall let

$$F_t(\varepsilon | k) = P(i_t < \varepsilon | \sigma_t = k), \tag{3.12}$$

and consider the conditional characteristic function of i_t, given σ_t, which is

$$M_t(j\omega, k) = \int_{-\infty}^{\infty} e^{j\omega i_t} dF_t(i_t | k). \tag{3.13}$$

From our definition of the first order channel, we may express i_t recursively as

$$i_t = \lambda i_{t-1} + c a_{t-1} = \lambda i_{t-1} + c(\sigma_t - \sigma_{t-1}). \tag{3.14}$$

The conditional distribution may also be expressed recursively. Considering the event, $\{i_t < \varepsilon | \sigma_t = k\}$, it is clear that this event is equivalent to

$$\bigcup_{\ell=1}^{d} \{i_{t-1} < \frac{\varepsilon - c(k-\ell)}{\lambda}, \sigma_{t-1} = \ell, \sigma_t = k\}. \tag{3.15}$$

Our assumption that the DS completely specifies the distribution of the I.I. at any time allows us to write the distribution function as

$$F_t(\varepsilon | k) = \sum_{\ell=1}^{d} F_{t-1}\left(\frac{\varepsilon - c(k-\ell)}{\lambda} \,\middle|\, \ell\right) \rho_{k\ell} \tag{3.16}$$

where $\rho_{k\ell} = P[\sigma_{t-1} = \ell | \sigma_t = k]$, are the <u>reverse</u> one-step transition probabilities of the stationary DS process. The reverse transition probabilities of the DS process are found in terms of the forward transition probabilities as

$$\rho_k = q_{k\ell} \frac{P(\sigma=\ell)}{P(\sigma=k)} \quad . \tag{3.17}$$

We shall let $R = \|\rho_{i,j}\|$ denote the matrix of reverse transition probabilities.

Letting $\eta = [\varepsilon - c(k-1)]/\lambda$, we may express the conditional characteristic function in (3.13) as

$$M_t(j\omega, k) = \sum_{l=1}^{d} \rho_{kl} \int_{-\infty}^{\infty} e^{j\omega[\lambda\eta + c(k-1)]} dF_{t-1}(\eta \mid l) = \sum_{l=1}^{d} \rho_{kl} e^{j\omega c(k-1)} M_{t-1}(j\lambda\omega, l). \tag{3.18}$$

We may disregard the explicit time dependence because we are interested in the steady state whereby the functional equation for the conditional characteristic function is,

$$M(j\omega, k) = \sum_{l=1}^{d} \rho_{kl} e^{j\omega c(k-1)} M(j\lambda\omega, l). \tag{3.19}$$

We have mentioned previously that we will utilize the moments of the I.I. in the error probability calculation. From (3.19), we may obtain the conditional moments of the I.I. and DS via the well known relationship between the moments and the characteristic function for a random variable. Thus, we have (see, e.g., [18])

$$j^p E[i^p \mid k] = \sum_{l=1}^{d} \rho_{kl} \frac{d}{d\omega} \{e^{j\omega c(k-1)} M(j\lambda\omega, l)\}\Big|_{\omega=0} . \tag{3.20}$$

The derivatives of the right-hand side of (3.20) may be conveniently expressed as

$$[jc(k-1) + \frac{d}{d\omega}\Big|_{\omega=0}]^p M(j\lambda\omega, l) \tag{3.21}$$

where the quantity in parentheses is understood to be an operator. Furthermore,

$$\frac{d^p}{d\omega^p} M(j\lambda\omega \quad ,1)\Big|_{\omega=0} = j^p \lambda^p \mu_1^p \tag{3.22}$$

where $\mu_1^p = E[i^p | \sigma = 1]$ is the p^{th} moment of the conditional random variable $\{i | \sigma = 1\}$. Thus for any moment of the I.I., we obtain a system of linear equations as follows:

$$\mu_k^p = \sum_{l=1}^{d} \rho_{kl} \sum_{i=0}^{p} \binom{p}{i} [c(k-l)]^i \lambda^{p-i} \mu_l^{p-i}; \; k=1,\ldots,d. \tag{3.23}$$

We can cast (3.23) into matrix form by rearranging terms:

$$(\lambda^{-p} - \rho_{kk}) - \sum_{l \neq k} \rho_{kl} \mu_l^p = \sum_{l=1}^{d} \rho_{kl} \sum_{i=1}^{p} \binom{p}{i} [c(k-l)]^i \lambda^{-i} \mu_l^{p-i}. \tag{3.24}$$

Expression (3.24) is a system of linear equations in the p^{th} conditional moments. This system is in terms of μ_l^k ;k=0,...,p-1, hence, these moments may be calculated recursively. In matrix form we have

$$[R(\lambda^{-p})]^T \underline{\mu}^p = \underline{b}_p. \tag{3.25}$$

The matrix $R(\lambda^{-p})$ is a tridiagonal matrix in the ternary case. The structure of $R(\lambda^{-p})$ is explicitly shown below.

$$\begin{bmatrix} \lambda^{-p}-\rho_{11} & -\rho_{21} & & & \\ -\rho_{21} & \lambda^{-p}-\rho_{22} & -\rho_{23} & & \\ & & \ddots & & \\ & & & -\rho_{d,d-1} & \lambda^{-p}-\rho_{d,d} \end{bmatrix} \tag{3.26}$$

It should be noted that (3.25) has a unique solution for all DS transition matrices. If $R(\lambda^{-p})$ were singular, then λ^{-p} would be an eigenvalue of R. Since R is a stochastic matrix, however, all of its eigenvalues have modulus less than 1 [32]. Because $|\lambda| < 1$, λ^{-p} cannot be an eigenvalue of R. Thus, in order to find the conditional I.I. moments up to order p, the vector $\underline{b}_p$ is calculated from the right-hand side of (3.24) and the system of equations in (3.25) solved.

It is interesting to specialize (3.25) in order to obtain the mean values (p=1). In this case, we have

$$\underline{b}_1 = c \begin{bmatrix} -\rho_{21} \\ \rho_{12}-\rho_{32} \\ \vdots \\ \rho_{i-1,i}-\rho_{i+1,i} \\ \vdots \\ \rho_{d-1,d} \end{bmatrix} \tag{3.27}$$

By solving the equation

$$[R(\lambda^{-1})]^T \underline{\hat{\mu}}^1 = (1/c)\underline{b}_1 \tag{3.28}$$

where $\underline{\hat{\mu}} = (1/c)\,\underline{\mu}$, we see that the mean values are directly proportional to c.

Finally, in regard to the conditional moments, we should note the calculation of the central conditional moments, which we will utilize in the next chapter when evaluating the error probability for a simple decision feedback detector based on the DS process. The adjustment is straightforward. Letting

$$\hat{M}(j\omega,k) = \int_{-\infty}^{\infty} e^{j\omega(\xi - \underline{\mu}_k^1)} dF(\xi|\, k) = e^{-j\omega\mu_k^1} M(j\omega,k) \tag{3.29}$$

be the conditional I.I./DS characteristic function when the I.I. is translated by its mean value, we have

$$\hat{M}(j\omega,k) = \sum_{l=1}^{d} \rho_{kl} e^{j\omega\,[c(k-1)-\mu_k + \lambda\mu_l\,]} \hat{M}(j\lambda\omega,l). \tag{3.30}$$

Now the expression for the central moments becomes

$$\mu_k^p = \sum_{l=1}^{d} \rho_{kl}\{[c(k-1)-\mu_k + \lambda\mu_l\,] + \lambda\mu_l\}^p. \tag{3.31}$$

Thus, to find the central moments, we substitute

$$c(k-1) - \mu_k + \lambda\mu_l \tag{3.32}$$

for c(k-1) in the right-hand side of (3.24).

3.3 Evaluation of Error Probability

In section 3.1 we derived the expression for the error probability in terms of the conditional probabilities, (3.9), as

$$P(E)=\sum_{j=1}^{d} P(E \mid j)P(\sigma=j). \tag{3.33}$$

In the preceding section we derived the joint moments of the I.I. and DS. Because the I.I. and noise are independent, the conditional distribution for their sum is found from the convolution

$$\int_{-\infty}^{\varepsilon} \int_{x\varepsilon I_j} F_{N|\sigma}(\xi-x|j)dF_{I|\sigma}(x|j)d\xi . \tag{3.34}$$

where I_j is the range of the conditional I.I. random variable. The conditional noise distribution is $F_N(\cdot)$ since the noise is independent of the DS. Recalling that we have assumed the noise is Gaussian, we may express the conditional I.I. plus noise distribution as

$$F_{I+N|\sigma}(\varepsilon \mid j)=\int_{-\infty}^{\varepsilon} \int_{x\varepsilon I_j} \Phi(\xi-x)dF_{I\mid\sigma}(x|j)d\xi \tag{3.35}$$

where

$$\Phi(z)=\frac{1}{\sqrt{\pi N_0}} e^{-z^2/N_0}, \tag{3.36}$$

is the Gaussian density function with variance equal to $N_0/2$.

If we expand (3.35) into a Taylor series about the point ξ, we obtain

$$F_{I+N|\sigma}(\varepsilon|j)=\int_{-\infty}^{\varepsilon}\int_{x\varepsilon I_j}\sum_{\ell=0}^{\infty}\frac{(-x)}{\ell!}\Phi^{(\ell)}(\xi)\,dF_{I|\sigma}(x|j)\,d\xi \tag{3.37}$$

where $\Phi^{(\ell)}(\xi)$ denotes the ℓ^{th} derivative of $\Phi(\cdot)$ evaluated at the point ξ. Expression (3.37) can be rearranged to produce

$$\begin{aligned}\int_{-\infty}^{\varepsilon}\Phi(\xi)d\xi+\sum_{\ell=1}^{\infty}\frac{(-1)}{\ell!}\int_{-\infty}^{\varepsilon}\Phi^{(\ell)}(\xi)d\xi\int_{x\varepsilon I_j}x^{\ell}dF_{I|\sigma}(x|j)&=\\ \int_{-\infty}^{\varepsilon}\Phi(\xi)d\xi+\sum_{\ell=1}^{\infty}\frac{(-1)}{\ell!}\Phi^{(\ell-1)}(\varepsilon)\mu_j^{\ell}&.\end{aligned} \tag{3.38}$$

where μ_j^{ℓ} are the conditional I.I. moments. Before we calculate the total error probability, we must mention the evaluation of the derivatives of $\Phi(z)$. Fortunately, these may be evaluated recursively since (see, e.g., [15])

$$\Phi^{(\ell)}(\xi)=\left(-[2/N_0]\right)^{\ell/2}H_{\ell}[2\xi/N_0]\,e^{-\xi^2/N_0} \tag{3.39}$$

where H $(\cdot)$ is the Hermite polynomial of degree ℓ. The Hermite polynomials are recursively related as

$$H_{\ell}(z)=zH_{\ell-1}(z)-(\ell-1)H_{\ell-2}(z) \tag{3.40}$$

with the initial conditions $H_0(z)=1$ and $H_1(z)=z$. We may substitute (3.38) into (3.9) using (3.39) to find the conditional error probability. After some algebraic manipulation, we arrive at

$$P(E|\sigma=j) = (q_{j,j+1} + q_{j,j-1} + 2q_{j,j})Q\left(\frac{1}{\sqrt{2N_0}}\right)$$

$$+ \sum_{\ell=1}^{\infty} \frac{1}{\ell!\,\sqrt{2\pi}} [2/N_0]^{\ell/2} e^{-1/4N_0} \mu_j^{\ell}$$

$$\cdot \{H_{\ell-1}([2N_0]^{-\frac{1}{2}})(q_{j,j} + q_{j,j-1}) + H_{\ell-1}(-[2N_0]^{-\frac{1}{2}})(q_{j,j} + q_{j,j+1})\} \tag{3.41}$$

where

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^{\infty} e^{-\xi^2/2}\, d\xi \ . \tag{3.42}$$

Further simplification of (3.41) is possible by noting that

$$q_{j,j+1} + q_{j,j-1} + 2q_{j,j} = 1+q_{j,j} \tag{3.43}$$

In addition, we note that the Hermite polynomials have the following property

$$H_\ell(x) = (-1)^\ell H_\ell(-x). \tag{3.44}$$

In words, the Hermite polynomials of even order are even functions of their arguments, and those of odd order are odd functions. These considerations allow us to rewrite (3.41) as

$$P(E|\sigma=j) = (1 + q_{j,j})Q\left(\frac{1}{\sqrt{2N_0}}\right) + \sum_{\ell=1}^{\infty} \frac{1}{\ell!\,\sqrt{2\pi}} (2/N_0)^{\ell/2} H_{\ell-1}([2N_0]^{-\frac{1}{2}})$$

$$\cdot\, e^{-1/4N_0} \mu_j^{\ell}[(q_{j,j} + q_{j,j-1}) + (-1)^{\ell}(q_{j,j} + q_{j,j+1})]. \tag{3.45}$$

The total error probability is found to be

$$P(E) = (1 + P_0)\, Q\left(\frac{1}{\sqrt{2N_0}}\right)$$

$$+ \sum_{\ell=1}^{\infty} \frac{1}{\ell!\,\sqrt{2\pi}}\, e^{-1/4N_0}\, H_{\ell-1}([2N_0]^{-\frac{1}{2}}) \tag{3.46}$$

$$\cdot \{\sum_{j=1}^{d} \mu_j^{\ell}\, P(\sigma=j)[(q_{j,j} + q_{j,j-1}) + (-1)^{\ell}(q_{j,j} + q_{j,j+1})]\}$$

where P_0 denotes the probability that the channel symbol is zero. The first term in (3.46) is seen to be the error probability for the code when the I.I. is zero, and the infinite series is the effect on the error probability by the I.I.

In the next section we shall present some results of this analysis for several known line codes.

3.4 Illustration of the First Order Model of I.I.

We have developed a method to evaluate the error probability for a first order model of I.I. There are two parameters, c and λ, both of which can be varied to produce different conditions. The parameter c is the maximum of the error channel's impulse response. It indicates the magnitude of the I.I. relative to the magnitude of the signal. Thus we may adjust c to obtain a variety of signal-to-I.I. ratios. In addition, the sign of c is related to the type of channel in that a high pass channel will be modeled by a negative value of c, and a lowpass channel by a positive value. The parameter λ indicates the length of time for which one channel symbol interferes with its successors. If we consider the number of symbols for which the error channel response is greater than .1% of c as a criterion for the duration (denoted τ), we may find the corresponding value for λ as $10^{-3/\tau}$.

The first set of curves which we display represent an error channel with τ equal to 100 symbols (a slow response). The value of λ corresponding to this is .933. In Fig. 11 we show P(E) for the PST code for four values of c (viz. -.1,0,.05,.1). The zero value of c is equivalent to the zero I.I. case. It is seen that the negative value of c does not degrade P(E) as much as the equal but opposite positive value. In fact, for $c \in [-.05,-.005]$ we found a slight improvement in the error probability at low signal-to-noise ratios. An intuitive explanation for this is the similarity of the the DS random process to a random walk with reflecting barriers. As the DS increases towards a barrier, the I.I. tends to decrease (become more negative). The probability of the DS changing directions, however (i.e., the probability of a channel symbol with negative weight), increases so that the I.I. tends to enhance the noise margin (i.e., it adds constructively to the signal). When the SNR is low this

constructive I.I. lowers the error probability slightly. This also explains why the positive values of c increase the P(E) at low SNR since the I.I. tends to be destructive (degrades the noise margin) most of the time. We see from Fig. 11 that when $P(E)=10^{-5}$, the degradation in the SNR is approximately 2 db for the severe I.I. case (c=.1).

In Figs. 12 and 13 we show P(E) for several known codes, including our running example, PST. The other codes are the Bipolar[19], MS43[20], an (8,6,3) code, and a (2,2,3) code[21]. In these illustrations λ is again .933 and c is .1 (in Fig. 12) and -.1 (in Fig. 13). At this high level of I.I. there is a considerable difference in the error performance of different codes. For $P(E)=10^{-5}$ and c=-.1, the difference between the best and worst is approximately 2.6 db, while for c=.1, this difference is approximately 3.3 db.

Next we present an identical set of curves for λ equal to .501. This corresponds to a fast channel response ($\tau=10$). In Fig. 14, P(E) is plotted for the PST code with values of c corresponding to Fig. 11. The worst error occurs again for c=.1 which degrades the SNR by approximately 1.8 db at $P(E)=10^{-5}$. A careful comparison of Fig. 11 with Fig. 14 reveals that the difference in error performance between $\lambda=.501$ and $\lambda=.933$ is small for PST, the maximum being approximately .5 db when c=-.1. However, when we compare the performance of the different codes in Fig. 15 and Fig. 16, we find that the difference between the best and worst error performance is only .9 db when $\lambda=.501$. The error performance of MS43 and (8,6,3) is considerably improved by the fast response of the channel.

In order to illustrate the dependence of P(E) on the duration of the channel response, we have plotted the error probability of MS43 with c=.1 for several values of τ ranging from 5 symbols to 200 symbols (Fig. 17). Observe

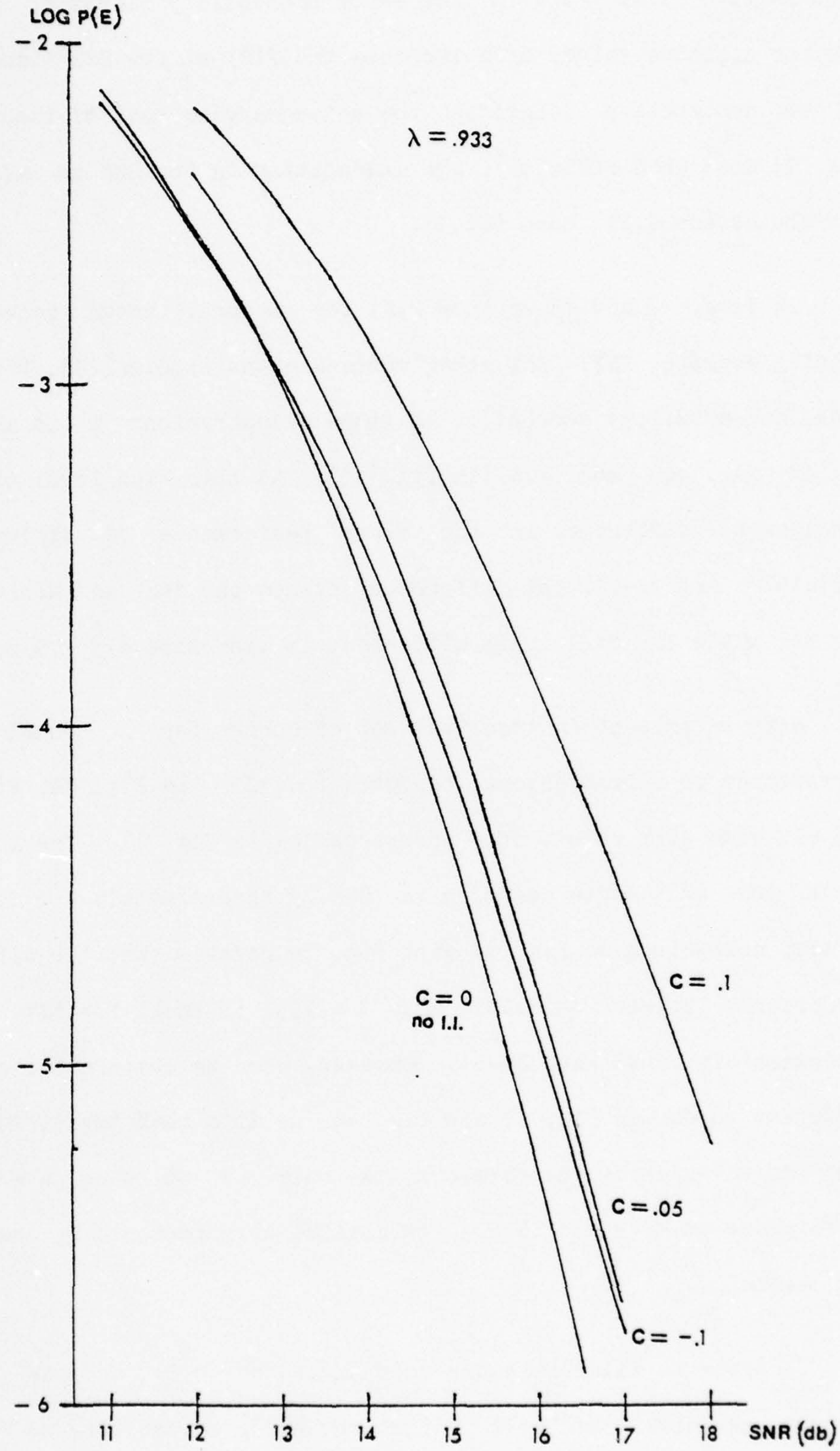


Figure 11. Error Probability for PST

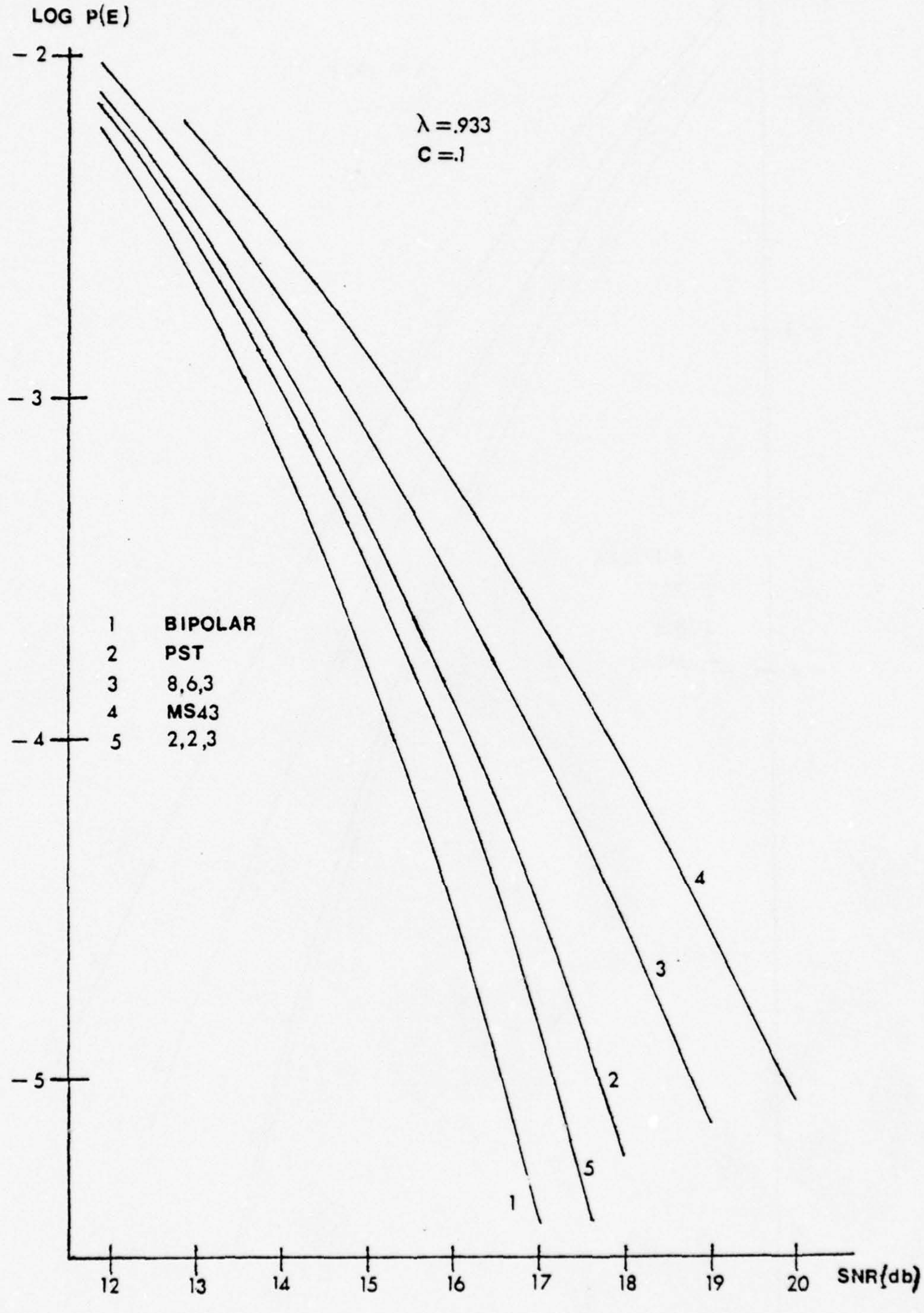


Figure 12. Comparison of P(E) for several codes (λ = .933; c=.1).

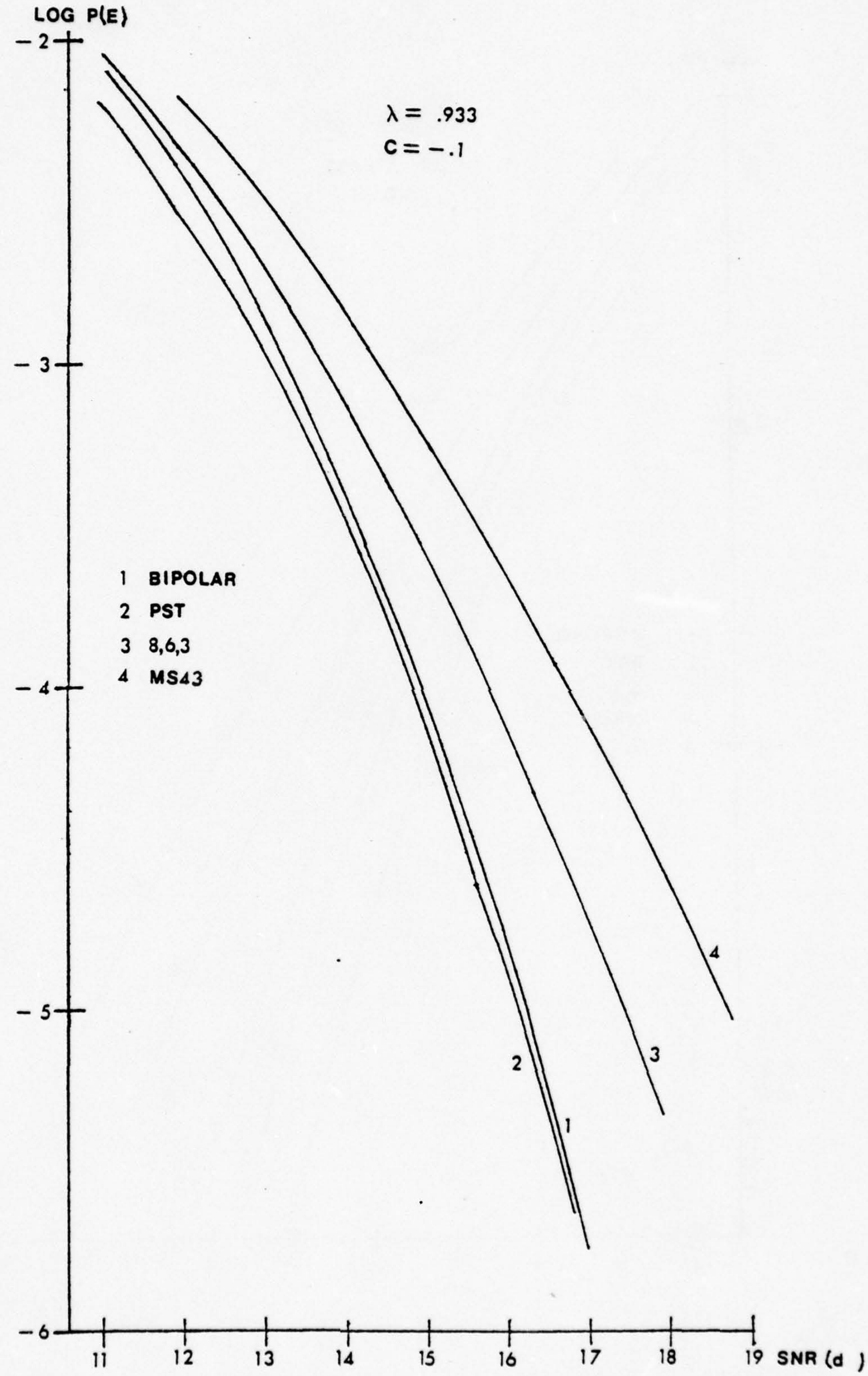


Figure 13. Comparison of P(E) for several codes (λ =.933; c=-.1)

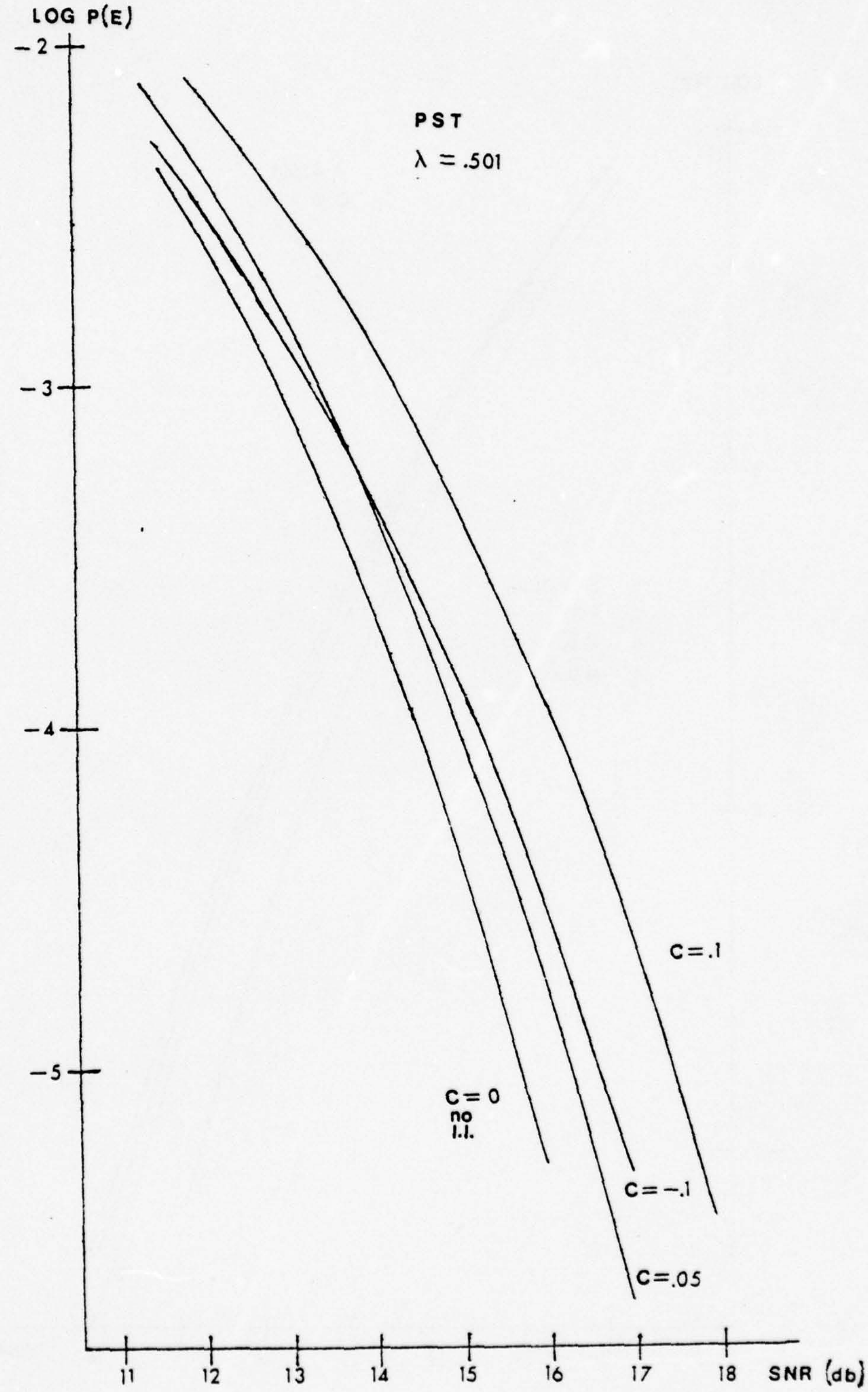


Figure 14. Error probability for PST (λ = .501)

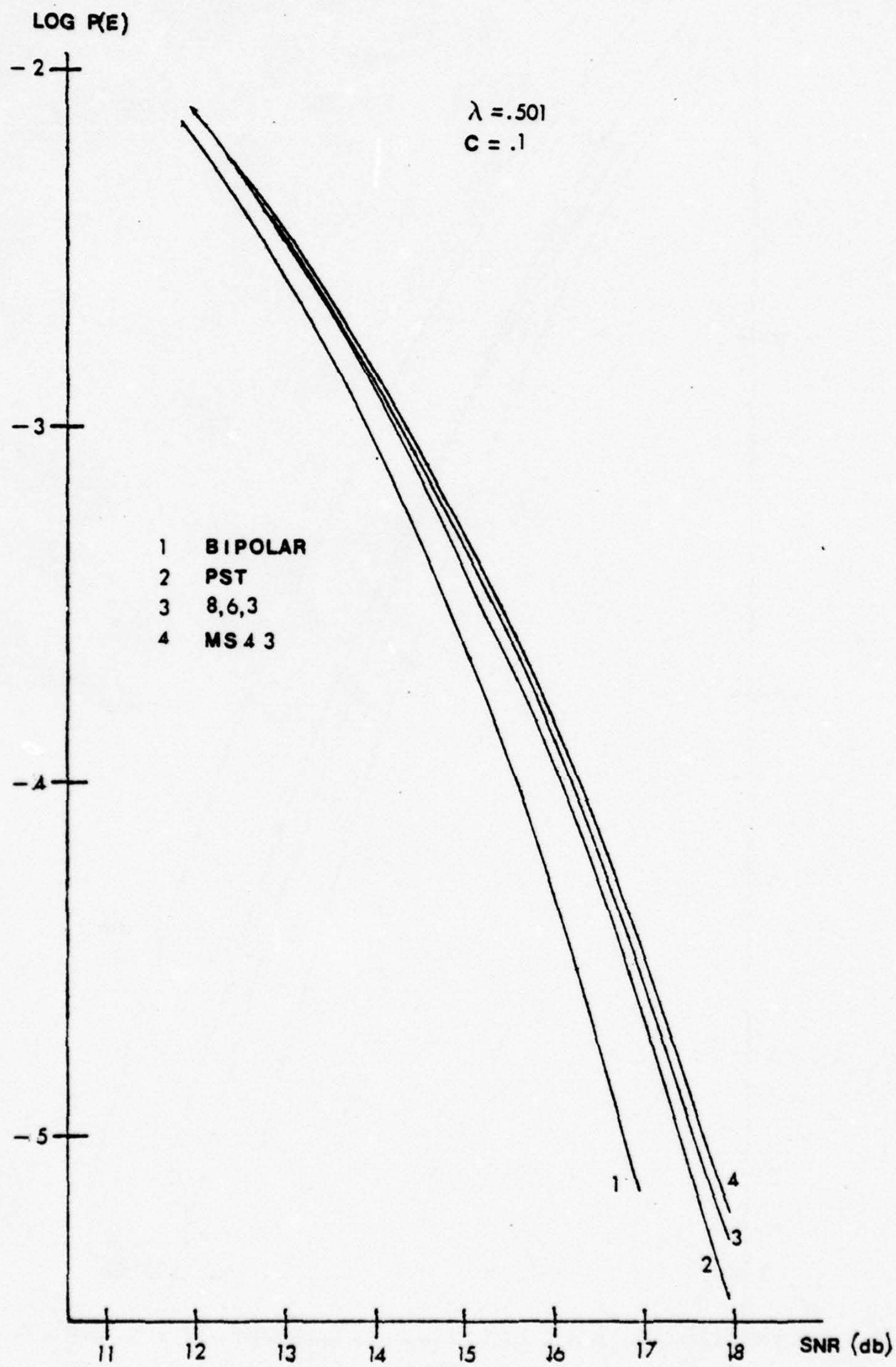


Figure 15. Comparison of P(E) for several codes (λ = .501; c=.1)

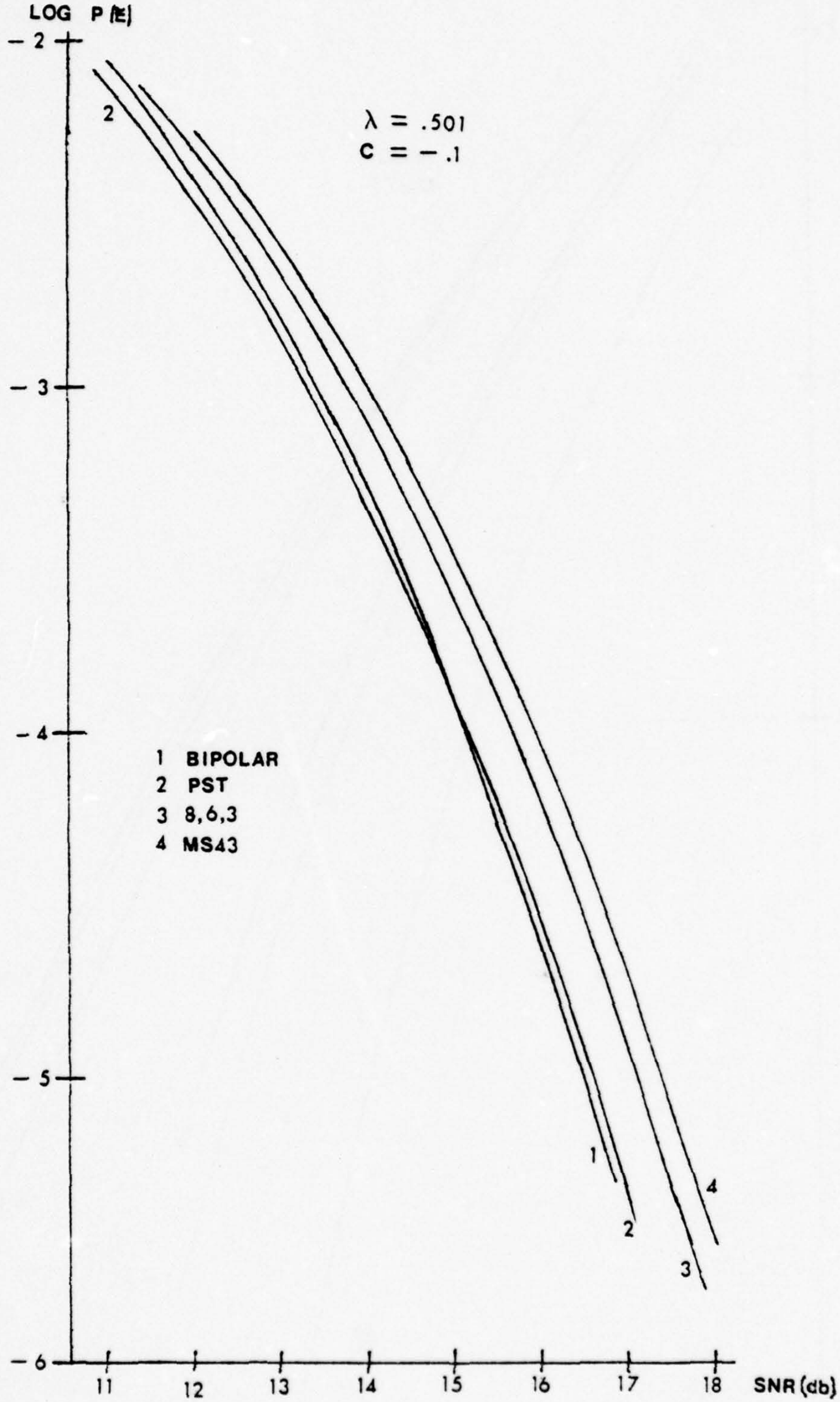


Figure 16. Comparison of P(E) for several codes (λ = .501; c=-.1)

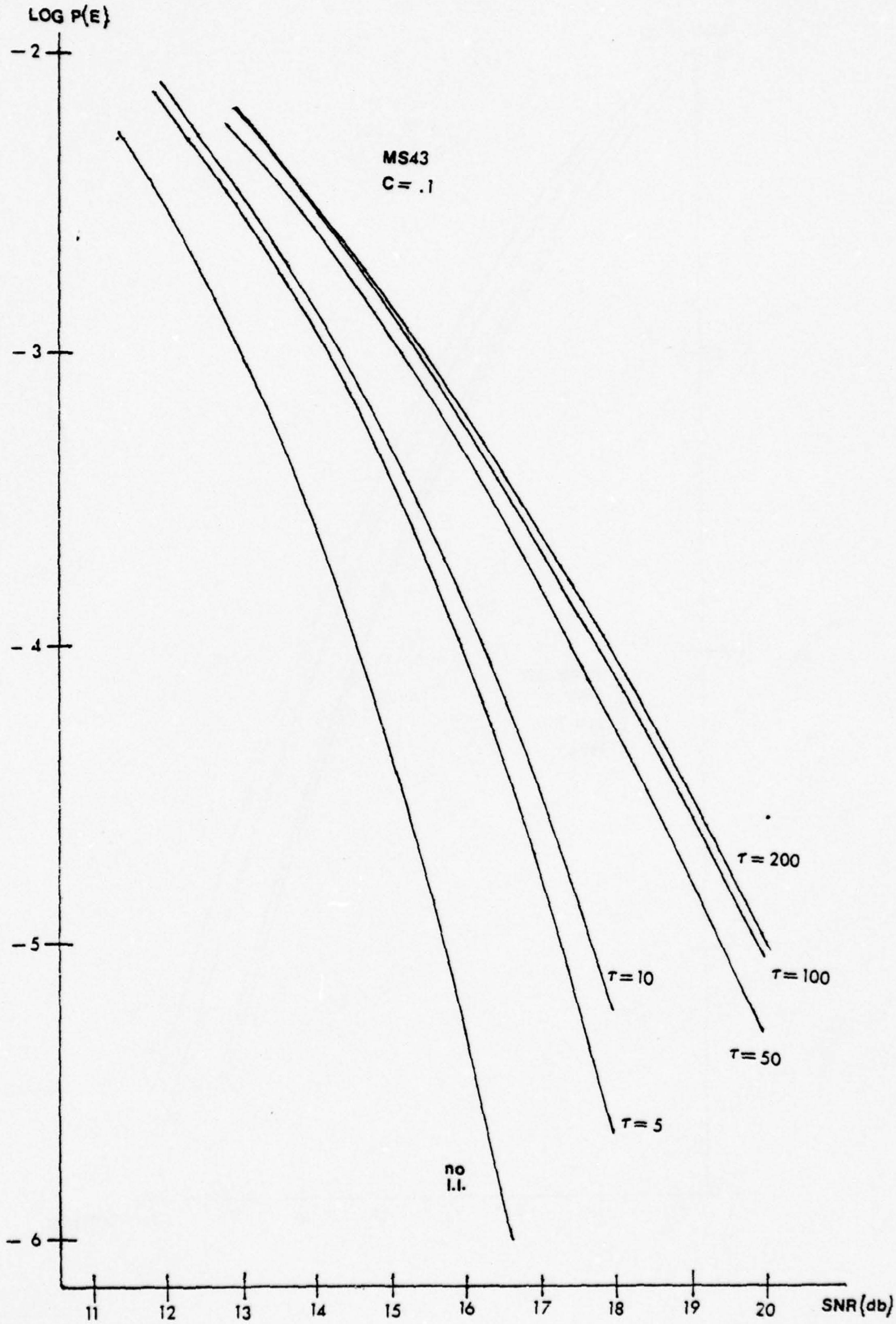


Figure 17. P(E) for MS43 as a function of response duration

that P(E) increases considerably as the duration, τ , increases from 10 to 50 symbols. However, for τ greater than 50 symbols (corresponding to $\lambda >$ .871), the difference is not great (approximately .6 db). In the next chapter, we shall investigate the behavior of the I.I. and P(E) for the case of a slow channel response.

CHAPTER 4

MODELS FOR SLOW CHANNELS

4.1 Digital Sum Channel

In the previous chapter, we have observed that when the first order error channel has an effective response of 50 symbols or greater, the parameter λ is close to unity. With this in mind, we shall consider a simplification of the first order model via its limiting behavior as λ approaches unity. Upon substituting $\lambda=1$ into the definition of the first order channel response (3.11), we obtain

$$\hat{h}_j = \begin{cases} 0; & j \leq 0 \\ c; & j \geq 1 \end{cases} \tag{4.1}$$

At first glance, it is easy to recognize that this system is unstable. For example, if a sequence of +1's of indefinite length were transmitted through this channel, the magnitude of the I.I. would increase indefinitely. But this property will not be a disadvantage for this analysis. The I.I. at some time k is

$$i_k = \sum_{j=-\infty}^{k-1} ca_j = c\,\sigma_k. \tag{4.2}$$

In words, the I.I. is directly proportional to the DS at every time. Because this analysis is limited to CE codes with a finite number of states, the DS will always be finite. Hence, even though the error channel (4.1) is unstable in the sense that

$$\sum_{j=0}^{\infty} |h_j| = \infty \quad , \tag{4.3}$$

it is stable for the restricted class of inputs under consideration. For convenience, we shall refer to this channel as the DS channel.

There is, however, one disadvantage in defining the I.I. for the DS channel as in (4.2). The values and statistics of the I.I. in this case are dependent on the choice of the initial value of the DS sequence. Intuitively we expect that the I.I. for any physical channel will be independent of this arbitrary specification. Indeed, we have found in the previous chapter that when λ is less than 1, the solution for the conditional moments in the first order model is unique regardless of this specification. Since we want to make the DS channel model as realistic as possible, it is necessary to to be more careful in our definition of the I.I. for this channel.

We may approach the definition of the I.I. for the DS channel by considering a property of CE codes. It should be recalled that we have assumed the channel symbol process is stationary with the statistics of the process derived from the arithmetic mean of the statistics of each phase of the cyclostationary codeword process. Furthermore, we know that the DS at any time is finite for a CE code with a finite number of states. Clearly, the expected value of the DS is finite. From the definition of the DS (2.15), we note that the expected value of the DS is

$$E(\sigma_k) = \sum_{j=-\infty}^{k-1} E(a_j) < \infty \quad . \tag{4.4}$$

From (4.4) we conclude that $E[a_j]=0$ for any CE code. Thus the I.I. for any

linear channel must have zero mean.

Thus far we have considered the DS values $\sigma(i), i=1,\ldots,d$ as integers for convenience. However, the only requirement we have placed on the DS values is

$$\sigma(i+1) - \sigma(i) = 1;\; i=1,\ldots,d-1. \tag{4.5}$$

The actual values of $\sigma(i)$ have been specified up to an additive constant. Prior to defining the I.I. for the DS channel, we must specify this additive constant from (4.4). The expectation of the DS via the ensemble average is

$$E(\sigma) = \sum_{i=1}^{d} \sigma(i)P(\sigma=i). \tag{4.6}$$

From (4.4) and (4.5), we obtain

$$E(\sigma) = \sum_{i=1}^{d} (C+i)P(\sigma=i). \tag{4.7}$$

Thus, the additive constant is found to be

$$C = -\sum_{i=1}^{d} iP(\sigma=i) \tag{4.8}$$

and we define the DS values as

$$\sigma(i) = C+i;\; i=1,\ldots,d. \tag{4.9}$$

This definition of the DS values assures that the definition of the I.I. for the DS channel is consistent with the limiting behavior of the first order model of I.I. as λ approaches unity.

The error probability for the DS channel is not difficult to compute. Since the DS sequence takes on d possible values, and the I.I. is directly proportional to the DS, the I.I. takes on values in the finite set,

$$\{\, c\,\sigma(j) \mid j=1,\ldots,d\}. \tag{4.10}$$

The probability density of the I.I. contains d impulses, which may be described as

$$\sum_{i=1}^{d} P(\sigma=j)\,\delta[i-c\,\sigma(j)]. \tag{4.11}$$

The conditional density of the I.I., therefore, is the impulse $\delta(i-c\sigma(j))$. By convolving this impulse with the Gaussian density function, we obtain the conditional density for the sum of the noise and I.I. for the DS channel as

$$f_{I+N|\sigma}(\xi \mid j)=\frac{1}{\sqrt{\pi N_0}}\,e^{-[\xi-c\,\sigma(j)]^2/N_0}. \tag{4.12}$$

From (3.9) we find the conditional probability of error as

$$\begin{aligned} P(E|j) = {} & (q_{j,j}+q_{j,j+1})\,Q\left(\frac{1+2c\,\sigma(j)}{\sqrt{2N_0}}\right)+ \\ & (q_{j,j}+q_{j,j-1})\,Q\left(\frac{1-2c\sigma(j)}{\sqrt{2N_0}}\right). \end{aligned} \tag{4.13}$$

The total error probability is found by averaging over the marginal events.

The error probability for the various codes discussed has been computed for the DS channel. The results are illustrated in Fig. 18 with c equal to .1. The comparison of Fig. 18 with Fig. 12 (λ =.933, c=.1) in Chapter 3 reveals that the

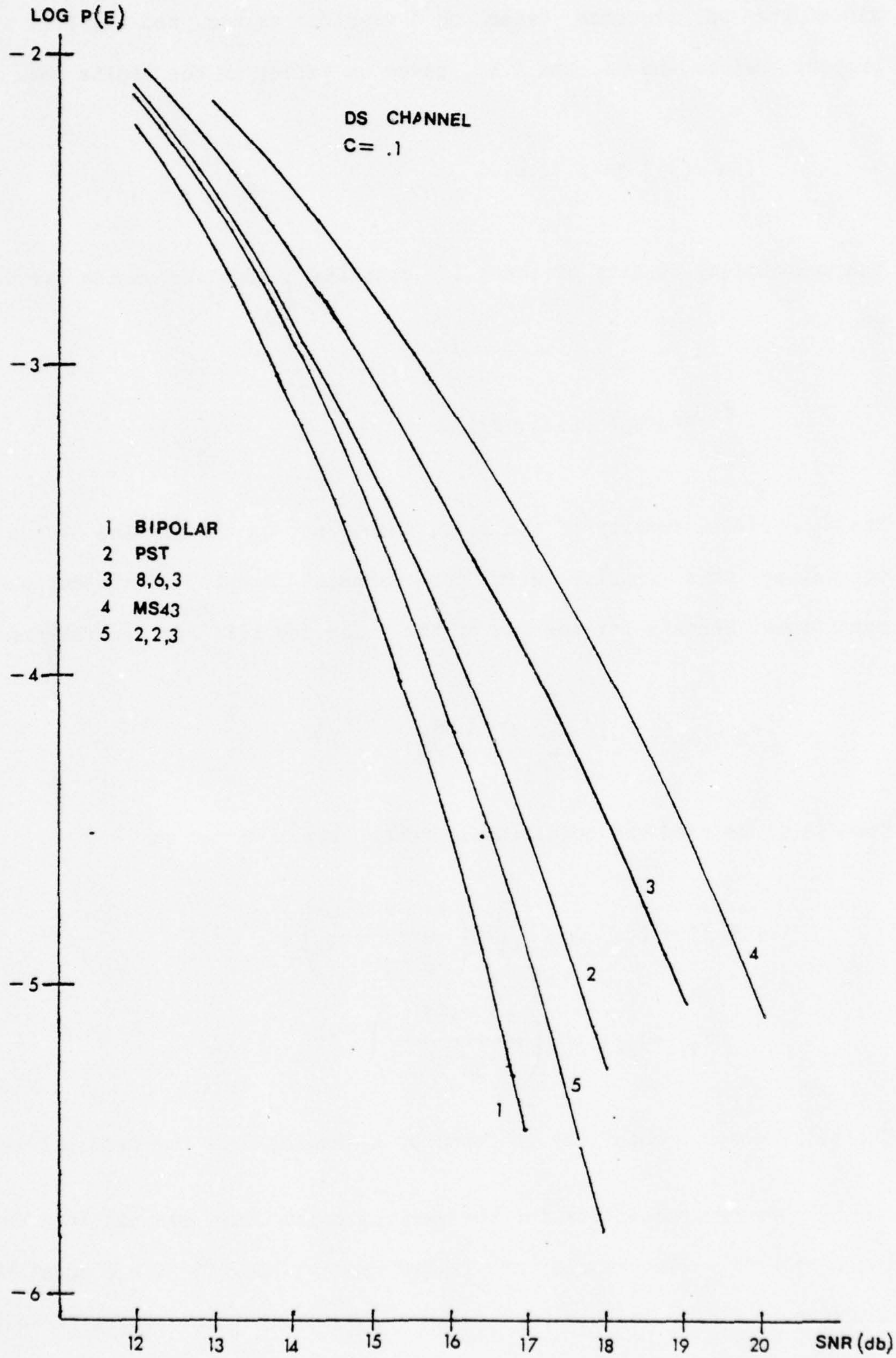


Figure 18. P(E) for several codes on the DS channel (c=.1)

difference in these two sets of curves is negligible. For these codes at least, the DS channel is an excellent approximation to a channel with a slow response. Comparing the error probability curve for the MS43 code in Fig. 18 with the set of curves shown in Fig. 17, indicates that the difference between an effective channel response of 50 symbols ($\lambda = .871$) and the DS channel is approximately .5 db for an error rate of 10^{-5}. Furthermore, among the codes discussed, the error probability for the MS43 has been found to be the most sensitive to the channel response duration.

At this point it is appropriate to discuss a simple parameter which appears in the literature as a criterion for the error performance for codes with bounded DS with respect to I.I. [7,22,23]. This quantity is called the Digital Sum Variation (DSV), which is defined as one less than the number of possible DS values that an encoded sequence may assume. Since we have consistently denoted the number of DS values as d; the DSV in this discussion is defined as d-1. The rule of thumb for judging a code's performance is: the smaller the DSV, the better the code. In [22], Croisier provided the motivation for the use of the DSV as a performance index by assuming that the I.I. is directly proportional to the DS. This assumption was made after a brief heuristic argument concerning the effects of a first order low pass filter on an encoded sequence. The results in this thesis thus far have hopefully provided an analytical basis for this assumption. In fact, the motivation for the DSV criterion is the assumption that the DS channel, as defined in this section, yields a good approximation to the error probability for channels with a slow response. Moreover, the DSV criterion is essentially the peak distortion criterion for the DS channel, and as such, it can be used in defining an upper bound for the error probability for the DS channel.

The DSV criterion can lead to erroneous conclusions when comparing code performance, however. In Table 4 , we present the DSV for each of the codes discussed. Upon comparing the DSV and the error performance illustrated in Fig. 18, we find that the criterion would lead to an incorrect evaluation of the MS43 code as superior to the (8,6,3).

Table 4. Digital Sum Variations

CODE	DSV
BIPOLAR	1
(2,2,3)	2
PST	3
MS43	5
(8,6,3)	6

A similar discrepancy has appeared in [24]. Although the actual error performance was not illustrated, it was noted that for the B6ZS code [25], the extreme values of the DS always occur so that the I.I. adds constructively to the next pulse, thereby enhancing the margin against the noise. This conclusion was made assuming a low frequency cutoff channel (which is equivalent to assuming c is negative in the DS channel model). The (8,6,3) code also has this property. Unfortunately, this line of reasoning does not explain why the (8,6,3) code provides a better error performance than the MS43 code when c is positive (which is the case in Fig. 18). In this case the extreme values of the DS always occur so that the I.I. adds destructively to the next pulse; thus reducing the margin against the noise. The next chapter will investigate these

questions in more detail when we propose techniques for the synthesis of CE codes for the DS channel. We shall find that the DS probability distribution, and the DS transition probabilities (or equivalently the joint DS/code-symbol distribution) can have a significant effect on the error performance of the code. When all of these factors are considered (including the DSV), we obtain a more accurate frame of reference for comparing code performance. Nevertheless, at this point, it is safe to say that a shadow of doubt has been cast on the DSV as an accurate indicator of the error performance of CE codes.

4.2 Decision Feedback Equalization for the DS Channel

Processing the decisions of the threshold detector in a feedback loop in order to cancel the I.I. is an old idea (see, e.g., [2]). In the past ten years this idea, now referred to as Decision Feedback Equalization (DFE), has received considerable attention. It has been found that significant improvement in performance can be obtained for low quality channels over the performance of linear equalization. Lucky provides a brief survey with an extensive bibliography on the subject in [5]. One problem with DFE is the propagation of errors in the detector decisions, which has led to the constraint of allowing only non-recursive (transversal) filters in the feedback loop, thus confining the error propagation to a finite number of future symbols. In this section we propose a very simple DFE scheme for the DS channel. Since the I.I. is directly proportional to the DS, the repeater can keep track of the DS after each decision by the threshold detector and compensate for the I.I. before a decision is made on the next symbol. Fig. 19 illustrates a model for this scheme.

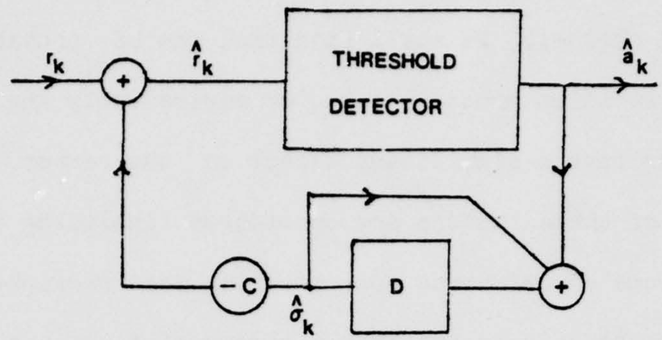


Figure 19. Model for digital sum feedback

Assuming no errors in the detector output up to time k, we have

$$\hat{r}_k = r_k - c\,\sigma_k = a_k + \xi_k. \tag{4.14}$$

In this case we see that, after compensation, the received signal is free of I.I. for the DS channel. Thus the error probability for this system is identical to the error probability for a system with no I.I.

As with any decision feedback scheme, the possibility of error propagation does exist. In fact, because the filter in the feedback loop is recursive (and unstable for unrestricted inputs), the error propagation could have infinite duration, and the magnitude of the error could become unbounded. We shall discuss how to avoid these problems later in this section, but we will first briefly mention the effects of a single error on the performance of this scheme.

If at some time k, the threshold detector's decision is in error (i.e., $\hat{a}_k \neq a_k$), the computed DS, $\hat{\sigma}_k$, will also be in error. In this case,

$$\hat{r}_k = a_k + \xi_k + c(\sigma_k - \hat{\sigma}_k). \tag{4.15}$$

The most common error is $a_k = a_k \pm 1$, that is, a channel symbol is detected as its adjacent symbol on the real line. *We shall refer to such an error as a single error*. Note that this is not the standard definition of a single error which appears in the coding literature. For a single error, (4.15) becomes

$$\hat{r}_k = a_k \pm c. \tag{4.16}$$

If σ_k is close to its maximum or minimum, we see that even with a single error, the compensated signal will be an improvement since most of the I.I. will be removed. If the DS is near zero, however, the compensated signal will be degraded, or at least not improved.

The fact that the DS for CE codes is a finite random variable allows for the possibility of error monitoring in the repeater. This monitor simply checks $\hat{\sigma}_k$ at every k to see that it is a valid DS value. If $\hat{\sigma}_k$ is not an allowable value, it increments a counter which keeps a running average of such DS violations. If this running average exceeds a predetermined threshold, the repeater may send an alarm signal to one of the terminals. A faulty repeater may be easily isolated in this manner. In the *Digital Sum Feedback* (DSF) scheme in Fig. 19 this monitor may be utilized to correct $\hat{\sigma}_k$ by the following rule:

$$\hat{\sigma}_k \leftarrow \begin{cases} \hat{\sigma}_k + 1; & \hat{\sigma}_k = \sigma_{min} - 1 \\ \hat{\sigma}_k - 1; & \hat{\sigma}_k = \sigma_{max} + 1 \end{cases} \tag{4.17}$$

When errors occur infrequently, single errors will be compensated for in the DSF device if the encoder reaches the appropriate extreme DS state before another error occurs. Even if a long burst of errors were to occur, the error in the estimated DS is overbounded by $\sigma_{max} - \sigma_{min}$.

For DSF equalization it is obvious that the Digital Sum Variation is irrelevant to the error probability when we consider the error free analysis. However, when we consider the equalizer with the DS correction device, the DSV has some redeeming value. When errors occur infrequently, one measure of the effectiveness of the correction device in the feedback loop is the average number of symbols until the appropriate terminal DS state is reached. The worst case occurs when σ_k is an extreme value and the symbol error at time k yields a DS estimate of its adjacent value. The output of the equalizer will not be corrected until the other extreme value is reached by the encoder. Consequently, the worst case average time until correction will depend on the DSV for the code in question.

4.3 Performance of DSF for the First Order I.I. Model

If we utilize the DSF scheme on channels which we model as first order with $\lambda<1$, it is clear that the I.I. is not entirely eliminated at each time since the variance of the conditional I.I. random variable is not zero (the density is not impulsive). Nevertheless, we would expect an improvement as a consequence of the removal of a strong bias in the I.I. (at least when the effective channel response is 50 symbols or more).

The error probability for such channels with DSF equalization is easily computed using the techniques discussed in the previous chapter. The received signal after compensation is

$$\hat{r}_k = a_k + i_k - \mu_k + \xi_k, \tag{4.18}$$

where μ_k is the conditional mean value of the I.I. for σ_k. We may consider the I.I. random variable to be

$$I_k = i_k - \mu_k \tag{4.19}$$

which is the translation of the conditional I.I. random variable by its mean. Thus the error probability may be computed from expression (3.46) by substituting the central conditional moments for the moments.

The improvement in the error probabilty which results from the DSF scheme is most dramatically illustrated in the case of the MS43 code. Fig. 20 shows P(E) for various values of the channel response duration, τ . The sensitivity to the duration has been reduced significantly. At an error rate of 10^{-5}, the greatest degradation in the signal-to-noise ratio relative to the zero I.I. case is approximately 1.3 db. In Chapter 3 we found a difference of 4.3 db, so that the DSF scheme has improved the SNR by 3 db. In addition, it is clear that for the DSF scheme the worst performance occurs for fast channel responses. This is expected because the DS model represents the limiting behavior of the model as the duration approaches infinity. We note, however, that a slight improvement is achieved even for the fast responses. This indicates that the DSF scheme may be useful in eliminating I.I. for the slow part of a channel response without increasing the deleterious effects the I.I. for the fast part.

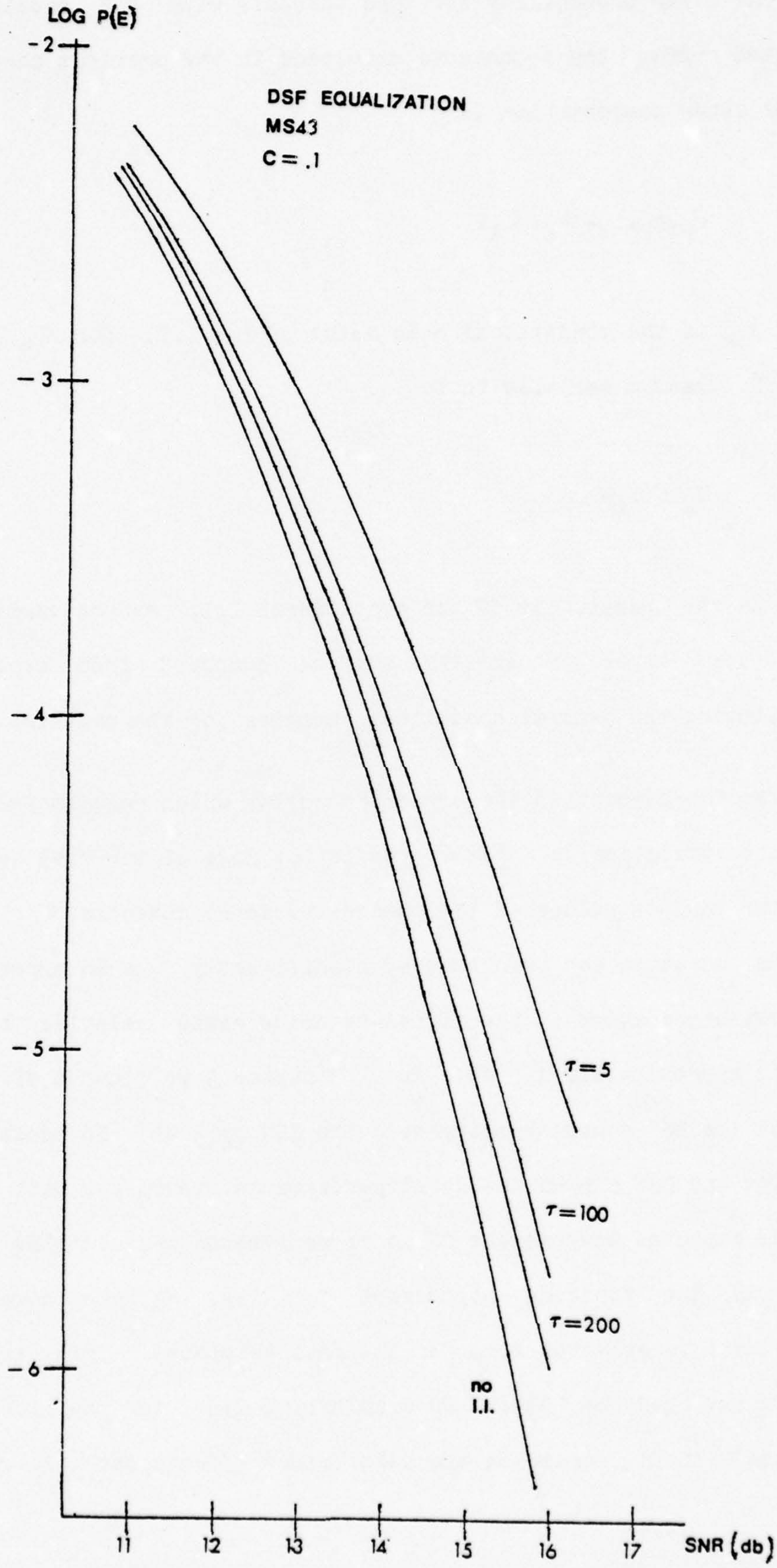


Figure 20. P(E) for MS43 with DSF Equalization

mode is "small," there are segments of the real line for which the probability of the I.I. being contained in these segments is essentially zero. Because the Gaussian function cannot approximate this behavior, the error probability obtained from it is overly conservative.

In this section, we shall propose the utilization of the Gaussian approximation for the <u>conditional</u> I.I. densities (i.e., we assume that the I.I. is Gaussian when conditioned on the DS). When λ is approximately unity we would expect this to yield a good approximation to the error probability since the conditional variances are small. In order to see this, consider the set of linear equations for the conditional variances:

$$\begin{aligned} &-\rho_{k,k-1}\sigma^2_{k-1}+(\lambda^{-2}-\rho_{k,k})-\rho_{k,k+1}\sigma^2_{k+1}= \\ &\frac{2}{\lambda^2}[\rho_{k,k-1}(c-\mu_k+\lambda\mu_{k-1})^2+\rho_{k,k}(-\mu_k+\lambda\mu_k)^2+\rho_{k,k+1}(-c-\mu_k+\lambda\mu_{k+1})^2] \end{aligned} \tag{4.20}$$

where σ^2_k denotes the conditional variances, and μ_k the conditional means. Recall that the conditional means are directly proportional to c and that in the limit as $\lambda \to 1$, the mean values are equally spaced with the separation being c. These considerations allow the rearrangement the right-hand side of (4.20), after which we obtain

$$\frac{2c^2}{\lambda^2}(1-\lambda)^2[\rho_{k,k-1}\hat{\mu}^2_{k-1}+\rho_{k,k}\hat{\mu}^2_k+\rho_{k,k+1}\hat{\mu}^2_{k+1}]. \tag{4.21}$$

Recalling that $(1-\lambda)$ is a factor of the characteristic polynomial of $R(\lambda^{-p})$, expression (4.21) allows us to state that the conditional variances are directly proportional to $(1-\lambda)^p$, where p is at least 1.

Another result which is not illustrated graphically is that the difference in P(E) for negative or positive c is negligible in all cases. Finally, it was also found that the difference in error performance of the various codes was reduced significantly with the DSF scheme. In Fig. 12 in the preceding chapter we found that the difference between the best and worst codes was approximately 3.3 db. for slow channels. With the DSF scheme we found a difference of less than 1 db.

4.4 Approximating the Probability Density of the I.I.

We have found the DS channel to be a reasonable approximation to the first order I.I. model with a slow response. This model indicates that the probability density of the I.I. is multimodal. It is interesting to note that this multimodal property appears in [2, p. 82] in an example of a channel with an exponential frequency response with raised cosine pulse shaping and an independent equiprobable binary channel sequence. The density in this case was found by the exhaustive method for the truncated time response.

It is desirable to find a good approximation for the I.I. density when the parameter λ is less than 1. It is well known that assuming the I.I. as Gaussian results in a poor approximation to the error probability. The explanation for this has been based on the fact that for any realistic channel the impulse response decays exponentially. The maximum of the I.I., therefore, is finite and the "tail" of its probability density cannot be Gaussian (see, e.g., [2]). The multimodal character of the I.I. density, however, suggests a stronger argument against the Gaussian approximation. When the variance of each

As λ approaches unity, the conditional density function becomes narrower until it "collapses" into an impulsive density. We may assume this density to be Gaussian with the mean value equal to the conditional mean, and variance equal to the conditional variance. This approximation is convenient because the density of the I.I. plus noise is also Gaussian, with mean and variance equal to the sum of the noise and I.I. means and variances, respectively. The conditional error probability is then

$$(q_{j,j}+q_{j,j+1})\int_{-\infty}^{-1/2}(2\pi\Gamma_j^2)^{-1}e^{-(\xi-\mu_j)^2/2\Gamma_j^2}\,d\xi$$
$$(q_{j,j}+q_{j,j-1})\int_{1/2}^{\infty}(2\pi\Gamma_j^2)^{-1}e^{-(\xi-\mu_j)^2/2\Gamma_j^2}\,d\xi \tag{4.22}$$

where $\Gamma_j^2=(N_0/2)+\sigma_j^2$ is the conditional variance of the noise plus I.I. After changing the variable of integration and averaging over the marginal events we obtain the total error probability as

$$P(E)=\sum_{j=1}^{d}P(\sigma=j)\left\{(q_{j,j}+q_{j,j+1})Q\left(\frac{1+2\mu_j}{2\Gamma_j}\right)+\right.$$
$$\left.(q_{j,j}+q_{j,j-1})Q\left(\frac{1-2\mu_j}{2\Gamma_j}\right)\right\}. \tag{4.23}$$

We shall also use this approximation for the DSF equalization scheme discussed in this chapter. In this case the combined I.I. and noise is zero mean Gaussian with variance Γ_j^2 , and the error probability is

$$\sum_{j=1}^{d}P(\sigma=j)(1+q_{j,j})Q\left(1/2\Gamma_j\right) \tag{4.24}$$

The accuracy of this approximation was tested on the codes discussed. It was found that the error probabilities resulting from the approximation agreed with the true error probabilities to two significant digits when the effective channel response was 50 symbols or greater. The maximum SNR for which these comparisons were made was 18 db. The greatest discrepancy between the approximate and the true P(E) occurred with the MS43 code when the effective response was only 5 symbols. This difference is approximately 1 db.

A final note on the accuracy of the Gaussian approximation for the conditional I.I. densities concerns the asymptotic behavior of the error probability as the SNR approaches infinity. The actual limiting behavior may be determined by assuming the channel to be noiseless. In this case, the error probability results from the I.I. only. The conditional error probability is a special case of expression (3.9). We find

$$\lim_{SNR\to\infty} P(E|j)=F_{I|\sigma}(-1/2|j)(q_{j,j}+q_{j,j+1})+\overline{F}_{I|\sigma}(1/2|j)(q_{j,j}+q_{j,j-1}). \tag{4.25}$$

If we consider the DS channel model, we find that the conditional I.I. probabilities are

$$F_{I|\sigma}(-1/2|j)=\begin{cases}1; & c\sigma(j)\leq -1/2\\ 0; & \text{otherwise}\end{cases} \tag{4.26}$$

and

$$\overline{F_I}\,|_\sigma(1/2|\,j) = \begin{cases} 1; & c\sigma(j) \geq 1/2 \\ 0; & \text{otherwise.} \end{cases} \qquad (4.27)$$

The limiting probability will, therefore, be zero if and only if $\sigma_{max} < 1/2c$. When we use the Gaussian approximation for $\lambda < 1$, the noiseless error probability will always be non-zero. However, the actual limiting noiseless error probability may be zero. When the conditional variances are small (λ close to unity), this difference should not be significant.

4.5 Results of Simulation: Conditional I.I. Densities

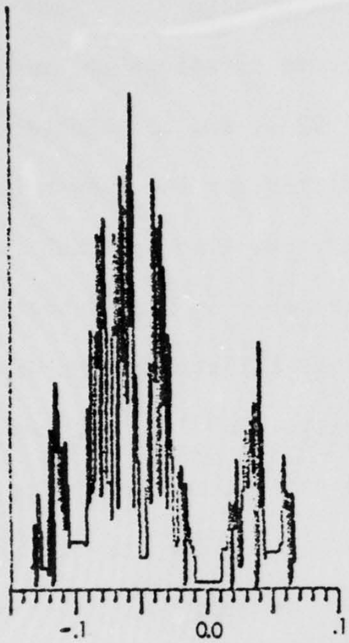
In this section we shall present some results of simulation of the first order I.I. model. The simulation method is a straightforward model of the encoder and first order channel. Because the input binary sequence was generated by a uniform random number generator, this sequence consisted of independent equiprobable random variables. The initial state of the encoder was selected arbitrarily. At each symbol time, the DS was computed and the I.I. value was stored in an array specified by the value of the DS. At the end of the simulation, the I.I. data for each value of the DS were sorted and quantized into intervals of width .001. We calculated the probability that the conditional I.I. value lies in a given interval by dividing the number of occurrences of the event by the total number of occurrences of the appropriate DS value.

First we shall present the "density" obtained for the Bipolar code, when $\lambda=.5$ and $c=.1$. It is known that the total I.I. density in this case is uniform [8,26]. The results shown are from a 90012 symbol simulation: the density for $\sigma=0$ is shown below in Fig. 21, and the density for $\sigma=1$ is identical, except for translation on the horizontal axis.

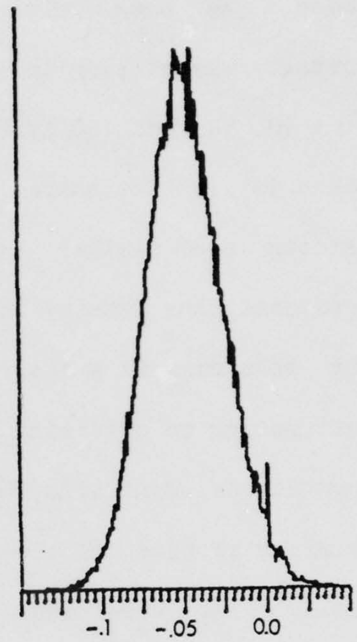
Figure 21. Conditional I.I. density for Bipolar

The conditional density shown in Fig. 21 is a reasonable approximation to the uniform density. Although we are making no claims about the statistical significance of the simulation results, we can be assured that the densities we obtain are a useful estimate of the actual densities. The results provide evidence for the accuracy of the Gaussian approximation for the conditional I.I. densities.

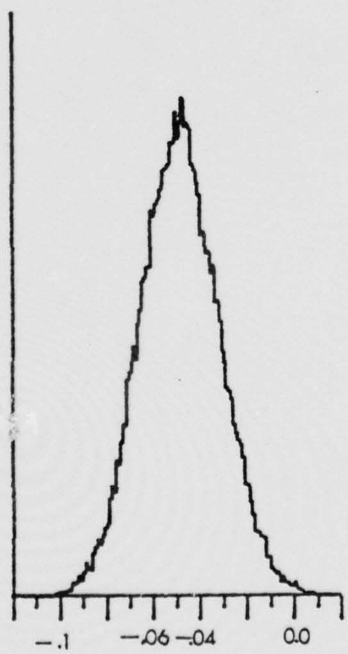
Fig. 22 illustrates some of the conditonal densities found for the PST code. We show only those for the DS equal to zero since each conditional density was similar in shape. The channel response durations we show are for τ= 10, 50, and 100 symbols (λ=.501,.871, and .933), and the constant c in each case is .1. The shapes of the conditional densities for the slow responses are smooth and similar to the Gaussian density. We also see that for the fast response, the density is not smooth. It is reasonable to assume that the I.I. in this case is a discrete random variable. This illustrates the difficulty of attempting to determine the densities analytically. In Fig. 23, we show the conditional densities for the MS43 code for the same parameters. The shapes are similar to those of the PST. This was found to be true for the other codes as well. The densities for the MS43 are not as smooth as those of the PST, however, which is probably a result of fewer data points in the simulation. The significance of the "spikes" at zero in the MS43 densities is also difficult to determine.



(a) 10 symbol duration

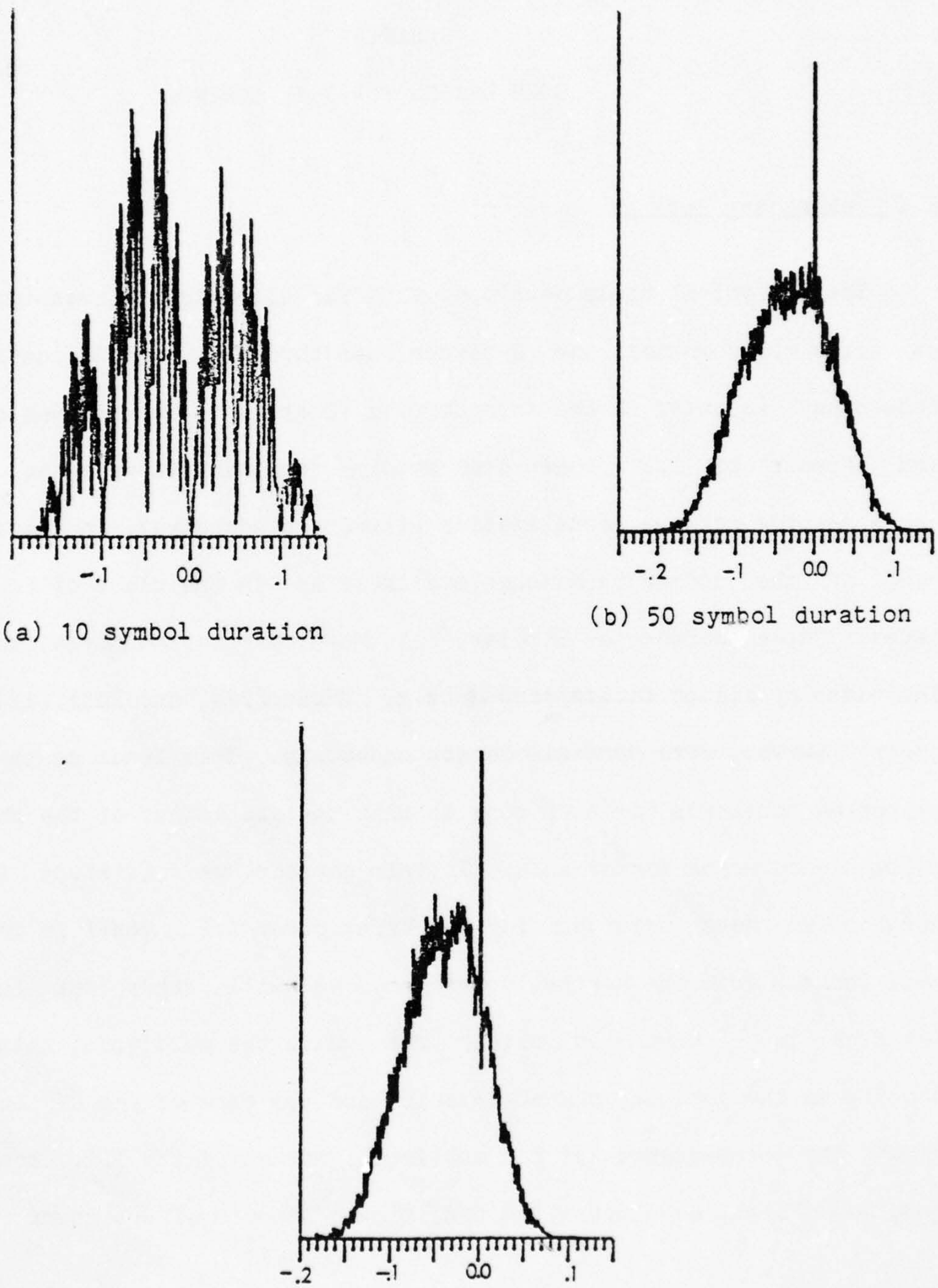


(b) 50 symbol duration



(c) 100 symbol duration

Figure 22. Conditional densities of PST for various response durations



(a) 10 symbol duration

(b) 50 symbol duration

(c) 100 symbol duration

Figure 23. Conditional densities for MS43 for various response durations

CHAPTER 5

CODE DESIGN FOR I.I. CONTROL

5.1 Preliminary Remarks

The analytical tools developed thus far allow us to investigate the impact on error performance the designer has through the judicious selection of CE codebooks. As noted in the introduction (Chapter 1), techniques for minimizing the effects of I.I. have been studied from many viewpoints; providing many pulse shaping schemes, equalization (fixed and adaptive), and coding techniques. Many of the coding techniques available are in the class of counter-encodable codes. These include the Bipolar, PST, MS43, partial response, and 4B-3T [27]. The class of filled bipolar codes [e.g. CHDBn [22], and BnZs [25]), are also CE codes, however, with variable length codebooks. This leads us to the problem of selecting codewords for a CE code so that the statistics of the channel sequence produce good error performance. In this chapter, we will study the analytical results we have obtained for the first order I.I. model in order to extract some insight into the synthesis problem. We will study the case where the detector is a threshold slicer for which the multimodal nature of the I.I. density is the dominant characteristic, and the case of the DSF equalization for which the consequences of the multimodal nature of the I.I. density have been removed. First, we discuss the preliminary aspects of designing CE codes.

5.2 Constraining Factors for CE Codebooks

Before we may concentrate on choosing the set of codewords which have the most desirable I.I. properties, it is necessary to determine the constraints which the structure of CE codes place on these sets. Recall that a CE code is identified as an (m,n,r) code where n is the codeword length (ternary), m is the blocklength of the binary information sequence and r is the number of states in the encoder. The first constraint follows directly from the definition of the state transitions for a CE code. Recall that the state transition function is

$$\sigma(s_{t+1}) = \sigma(s_t) + \gamma(\underline{a}_t) \qquad (5.1)$$

where $\gamma(\underline{a}_t)$ is the algebraic sum of the codeletters of the codeword at wordtime t. Since the set of states is finite and σ is a 1-1 function there is a least integer $\sigma(i)$ for some state i. When the encoder is in this state i, we have the restriction

$$\gamma(\underline{a}) \geq 0 \qquad (5.2)$$

for any $\underline{a}$ which is in state i. We refer to the codebook for this state as the <u>positive reflecting mapping</u> for the CE code. A similar argument demonstrates that there is another state, say j, such that $\gamma(\underline{a}) \leq 0$ for every $\underline{a}$ in state j. The codebook for this state is called the <u>negative reflecting mapping</u>. Since the set of states S is denoted by $\{1,2,\ldots,r\}$, we let state 1 be associated with the positive reflecting mapping and state r with the negative reflecting mapping. In this case we are assuming that the function σ is a strictly increasing function of the integers. For the majority of situations we consider, the function σ is $\sigma(i)=i$. The existence of the positive reflecting

mapping and the necessity of providing 2^m codewords in each state leads to a necessary condition for the existence of an (m,n,r) code. Let C_j denote the number of ternary n-tuples which have characteristic j, then

$$\sum_{j=0}^{n} C_j \geqslant 2^m \tag{5.3}$$

is this necessary condition. To demonstrate the sufficiency of (5.3), we must enumerate the C_j's. Toward this end, note that the concatenation of two codewords having lengths n_1 and n_2, and characteristics γ_1 and γ_2 respectively, yields a codeword having length n_1+n_2 and characteristic $\gamma_1+\gamma_2$. This suggests the generating function $(x^{-1}+1+x^1)^n$ as the characteristic enumerator, with the exponents indicating the characteristics and the coefficients being the number of words. Another restriction we place on the codebook is that the n-tuple of all zeroes be ineligible. The motivation for this restriction lies in the engineering of the regenerative repeaters. These repeaters derive the timing information for the sampling operation from the received sequence. If a long sequence of 0's occurs, there would be no timing information during this interval, and the sampler could lose symbol synchronization. Thus we must subtract 1 from the constant term in the enumerator, after which we obtain

$$(x^{-1} + 1 + x)^n - 1 = \sum_{j=-n}^{n} C_j x^j. \tag{5.4}$$

The symmetry of the enumerator implies $C_j = C_{-j}$ for every $j=1,\ldots,n$. Furthermore, ignoring the -1 term for a moment, it is clear that $C_j > C_k$ if $j < k$. Including the term -1 forces the inequality to be weakened to

$$C_j \geq C_k \text{ for } |j|<|k|. \tag{5.5}$$

Now, let us consider the encoding mode (mapping) for state k, $1 < k < r$. If $\underline{a}$ is a codeword in this mode, the state transition function requires

$$1-k \leq \gamma(\underline{a}) \leq r-k. \tag{5.6}$$

Let A_k denote the set of codewords satisfying (5.6), which we shall refer to as the set of k-eligible codewords (e.g., the set of codewords satisfying (5.2) is the set of 1-eligible codewords, A_1). From the definition of the set A_k, we see that

$$|A_k| = \sum_{1-k}^{r-k} C_j. \tag{5.7}$$

Inequality (5.5) ensures that $|A_k| \geq |A_1|$, therefore the condition (5.3) which is also equivalent to $|A_1| \geq 2^m$, is a necessary and sufficient condition for the existence of an (m,n,r) CE code. Table 5 shows the results of this enumeration for blocklengths up to seven.

Table 5. Enumerator Coefficients for $n \leq 7$

γ / n	0	1	2	3	4	5	6	7
1	1	1						
2	2	2	1					
3	6	6	3	1				
4	18	16	10	4	1			
5	50	45	30	15	5	1		
6	140	126	90	50	21	6	1	
7	392	357	266	161	77	28	7	1

Obviously, for a fixed codeword length, n, it is most desirable to maximize the binary blocklength, m. From the condition (5.3), it is a simple matter to find the maximum value of m for a given codelength. In addition, from the standpoint of hardware complexity, it is desirable to minimize the number of states, r. For any pair (m,n) the minimum number of states is the smallest r^* for which

$$\sum_{j=0}^{r^*-1} |A_j| \geq 2^m. \tag{5.8}$$

With the aid of Table 5, the code parameters yielding the greatest rate increases for codeword lengths up to seven have been determined with the minimum number of states necessary for a CE realization. Table 6 provides this information.

Table 6. Code parameters realizing greatest rate increase

n	m	r_{min}	Rate Increase
2	2	2	1
3	4	4	1.33
4	5	2	1.25
5	7	4	1.40
6	8	2	1.33
7	10	4	1.43

The code design problem is simplified by considering a sub-class of the general CE codes, which we refer to as balanced CE codes. A few conventions will be useful prior to the definition of these codes. A codeword $\bar{\underline{a}}$ is the complement of $\underline{a}$ if $-\bar{\underline{a}}=\underline{a}$ where the minus sign indicates the integer multiplication of each code digit by -1 (e.g., the complement of +0- is -0+). Moreover, the encoding mode, f_i, for some state i is balanced if $\underline{a} \in f_i \leftrightarrow \bar{\underline{a}} \in f_i$. Finally, two encoding modes, f_i and f_j, are complementary modes if $\underline{a} \in f_i \rightarrow \bar{\underline{a}} \in f_j$. A CE code is defined as a balanced code if every mode is either balanced or the complement of another mode; furthermore, if f_i is unbalanced then f_{r-i+1} is its complementary mode.

All of the CE codes which have been discussed in this thesis are balanced codes. Referring to the codebook for the PST code in Chapter 1 (Table 1), it is readily seen that the two states are complementary. The synthesis of balanced codes is somewhat easier than that of general CE codes because only half of the codewords must be selected. For example, if a 2-state code is designed, the two states must be complementary. Once the codewords for one state are chosen, the codewords for the other state are the complements.

The analysis of the DS process is also simplified for balanced codes. For every occurrence of $\sigma=i$, $i=1,\ldots,d$, in the DS table, there is an occurrence of $\sigma=d-i+1$ which implies $P(\sigma=i)=P(\sigma=d-i+1)$. Because the expected value of the DS is zero, the values of the DS are $\sigma(i)=i-(d+1)/2$; that is, the I.I. for the DS channel is symmetric about the origin. These considerations will be useful in the next section when we analyze the error probabilities for CE codes and develop some ideas concerning their synthesis.

There is another desirable property of CE codes to consider in this section. This property is called <u>State Independent Decodability</u> (SID). A CE code is SID if the decoder is able to map the received ternary n-tuple into the correct binary m-tuple without knowledge of the encoder's state. Of course this assumes that no transmission errors have occurred. Referring to the codebook, this definition is equivalent to saying that a ternary n-tuple may appear in at most one row for an SID code. A sufficient (but not necessary) condition for state independent decodability is (see [20], p. 147):

for any two states u,v such that $1 \leq u < v \leq r$, and a codeword $\underline{a}$;

$$\underline{a} \in u \cap v \rightarrow \underline{a} \in u+1. \tag{5.9}$$

In the next chapter, we shall discuss the practical consequences of decoding state-dependent or state-independent decodable codes. For the remainder of this chapter, we shall incorporate the SID property into the code synthesis problem for I.I. control.

The existence arguments presented in this section define the sets of codewords which are eligible for insertion into a CE code. Any synthesis algorithm thus begins by partitioning the set of 3^n-1 ternary n-tuples according to characteristic. Then the sets of k-eligible codewords may be formed for $k=1,\ldots,r$. The codebook may be filled by selecting n-tuples from these sets and inserting them into the appropriate mappings. Since the codes of interest are non-linear, we are free to choose the codewords by any rule we desire. The remainder of this chapter is concerned with developing selection rules based on the error performance of channel sequences for the DS channel.

5.3 Coding for Threshold Detection

In this section, we will use the DS channel model as the basis for determining the criteria to guide the code construction procedure. In the case that the detector is a fixed threshold device, we have found that the error probability is closely related to the multimodal nature of the I.I. probability density. From expression (4.13) we see that the P(E) is a function of the DS transition probabilities, the set of DS values, and the distortion coefficient, c, of the channel. Of these factors, the distortion coefficient is out of the code designers' control. However, some knowledge of this level is very useful, which we will demonstrate in this section. We shall begin by considering the limiting case of a noiseless system. This is a reasonable point of view because the systems of interest operate at a high SNR. Furthermore, the noiseless system provides insight into the noisy case. Let us define two sets as follows:

$$
\begin{aligned}
D^- &= \{j \mid \sigma(j) \leqslant -1/2c\}, \\
D^+ &= \{j \mid \sigma(j) \geqslant 1/2c\}.
\end{aligned}
\tag{5.10}
$$

In terms of these sets, we may express the noiseless error probability (denoted $P_\infty(E)$) from (4.25), (4.26), and (4.27) as

$$P_\infty(E) = \sum_{j \varepsilon D^-} P(\sigma=j)(q_{j,j}+q_{j,j+1}) + \sum_{j \varepsilon D^+} P(\sigma=j)(q_{j,j}+q_{j,j-1}). \tag{5.11}$$

In order to demonstrate the dependence of P (E) on the distortion level, we have plotted the limiting error probability in Fig. 24 for the PST code.

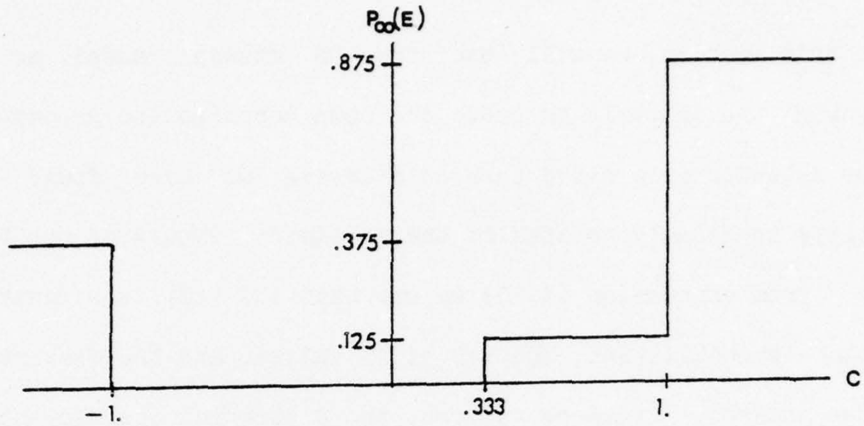


Figure 24. Noiseless error probability for PST

We see from Fig. 24 that when $0 < c < 1/3$, the noiseless error probability is zero. However, $P_\infty(E)$ jumps to .125 at c=1/3, and this error probability is unacceptable for any application. The values of c for which P (E) is discontinuous are a function of the DSV because the first point of discontinuity occurs when the DS is maximum. If we had no estimate for the distortion level for a particular application, it would be necessary to design a code with minimal DSV. However, this could be overly conservative because there is a penalty for minimizing the DSV. That is, when we design a code for minimal DSV, we must eliminate many codewords from consideration, and the rate (or efficiency) of the code is reduced. One design objective is the selection of the most efficient code for any application (i.e., one which maximizes m/n). However, as we have seen from Fig. 24, the magnitude of the distortion should be estimated first in order to determine an acceptable DSV.

Another aspect of the noiseless error probability is apparent from Fig. 24. That is, $P_\infty(E)$ is not symmetric about c=0. The error probability for negative c is less than that for positive c, and the range of c for which $P_\infty(E)$ is zero is

greater. We have noticed this before when the true error probabilities were calculated for the first order I.I. model, and again in the preceding chapter when the DSV was introduced. This asymmetry is easy to explain by assuming that $1/(d-1) < c < 1/(d-2)$ whereby $P_\infty(E)$ is nonzero for a code with d DS values, with the errors occurring only for the extreme DS values, $\sigma(1)$ and $\sigma(d)$. When c is positive, we have $D^-=\{1\}$, and $D^+=\{d\}$ which yields

$$P_\infty(E)=P(\sigma=1)(q_{1,1}+q_{1,2})+P(\sigma=d)(q_{d,d}+q_{d,d-1}). \tag{5.12}$$

When c is negative in the interval $(-1/(d-2),-1/(d-1))$, however, we find $D^-=\{d\}$ and $D^+=\{1\}$, whereby we obtain

$$P_\infty(E)=P(\sigma=d)(q_{d,d}+q_{d,d+1}) + P(\sigma=1)(q_{1,1}+q_{1,0}). \tag{5.13}$$

But because $\sigma=1$ and $\sigma=d$ are extreme states, $q_{d,d+1}=q_{1,0}=0$, and expression (5.13) becomes

$$P_\infty(E)=P(\sigma=d)\, q_{d,d}+P(\sigma=1)q_{1,1}. \tag{5.14}$$

We see that if $q_{d,d}=q_{1,1}=0$, the noiseless error probability is zero. This is the property we mentioned in the preceding chapter concerning the (8,6,3) code and the B6ZS code. From Fig. 24, it is apparent that the PST code also has this property. This demonstrates the weakness of using the DSV criterion as the only evidence in judging a code's performance.

For the case when $1/(d-1) < c < 1/(d-2)$, the expression for $P_\infty(E)$ may be simplified by noting for ternary codes; $q_{1,1}+q_{1,2}=q_{d,d}+q_{d,d-1}=1$. Thus, when only the extreme states are the cause of noiseless errors, the error probability

is

$$P(\sigma=1) + P(\sigma=d). \qquad (5.15)$$

If the probability in (5.15) could be made small enough so that an acceptable system performance is obtained, the requirement of $d-1 < 1/c$ could be relaxed. The minimum value of (5.15) for an (m,n,r) code occurs when there is only one appearance of $\sigma=1$ and $\sigma=d$ in the codes' DS table (see Chapter 1). We can obtain an estimate of this probability since the probability of one appearance of a symbol in the DS table is (see (2.19), and (2.20))

$$(1/n)p_i\ 2^{-m}. \qquad (5.16)$$

This assumes the input binary blocks are equiprobable. Since we only need a rough estimate of (5.15), we can assume that the stationary probability of the encoder state is unity, and that $n=m$. Thus an estimate of the least value of (5.15) is $(1/n)2^{-(n-1)}$. Using 10^{-5} as the maximum acceptable error probability, we find the required minimum codeword length is 14, which is too long for practical purposes. Therefore, for the case of the noiseless channel, we may assume that if $P_\infty(E)$ is not zero, then it is not acceptable on the DS channel.

The results from the analysis of the noiseless DS channel demonstrate that an estimate of the channel distortion coefficient may be useful in the code design process. For example, when $c=.1$, the noiseless error probability is zero when $d<11$. Practically speaking, this is an extremely large DSV, and is not a great restriction on the code design process. For the remainder of this thesis, we will assume that the noiseless error probability is zero for any code we discuss. In order to compare the performance of these codes, we must analyze

the noisy DS channel. We begin by considering the contribution to the error probability for a given DS value, which is $P(\sigma=j)P(E|j)$.

The first step is to determine the values of the DS which contribute the most to the error probability. In particular, we want to determine if the extreme values always contribute the most for a particular code (this is the tacit assumption of the DSV criterion). We need, therefore, a method to perform this comparison. A simplification which will not cause much inaccuracy in this analysis is to consider $q_{j,j}+q_{j,j+1}=q_{j,j}+q_{j,j-1}=1$. Recall that when $q_{1,1}=q_{d,d}=0$ and c is negative, the extreme sums do not contribute to the error probability. This is not a problem for this analysis because we may equivalently consider such a code as having σ_{max} equal to $(d-3)/2$ instead of $(d-1)/2$ when c is negative. With this simplification, we note that the contribution to P(E) for two DS values, j and k, will be equal if

$$\begin{aligned} &P(\sigma=j)\left\{Q\left(\frac{1+2c\,\sigma(j)}{\sqrt{2N_0}}\right)+Q\left(\frac{1-2c\sigma(j)}{\sqrt{2N_0}}\right)\right\}= \\ &P(\sigma=k)\left\{Q\left(\frac{1+2c\sigma(j)}{\sqrt{2N_0}}\right)+Q\left(\frac{1-2c\sigma(j)}{\sqrt{2N_0}}\right)\right\}. \end{aligned} \tag{5.17}$$

Next we note that the dominant term in the error probability, when c is positive, is

$$Q\left(\frac{1-2c\sigma(j)}{\sqrt{2N_0}}\right),$$

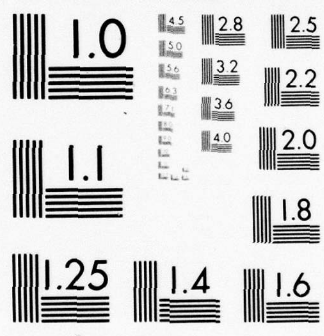
and when c is negative, the dominant term is

$$Q\left(\frac{1+2c\sigma(j)}{\sqrt{2N_0}}\right).$$

AD-A069 781 ILLINOIS UNIV AT URBANA-CHAMPAIGN COORDINATED SCIENCE LAB F/G 9/4
CODING FOR THE CONTROL OF INTERSYMBOL INTERFERENCE IN BASEBAND --ETC(U)
DEC 78 V A DIEULIIS DAAB07-72-C-0259
UNCLASSIFIED R-830 NL

2 OF 2
AD
A069781

END
DATE
FILMED
7 -79
DDC



MICROCOPY RESOLUTION TEST CHART

We may therefore consider only the dominant term in each side of (5.17) because the maximum error we could make is a factor of two which is not significant. Moreover, we may assume c is positive without loss of generality. These considerations allow us to rewrite (5.17) as

$$P(\sigma=j)Q\left(\frac{1-2c\sigma(j)}{\sqrt{2N_0}}\right)=P(\sigma=k)Q\left(\frac{1-2c\sigma(j)}{\sqrt{2N_0}}\right) \tag{5.18}$$

A useful tool which facilitates this analysis is a very tight upper bound for the Q-function [31], which is

$$Q(x) \leqslant (1/\sqrt{2\pi}\, x)\, e^{-x^2/2}. \tag{5.19}$$

Upon substituting (5.19) into (5.18) and rearranging the expression, we find that two DS values will contribute equally to the error probability if

$$\frac{P(\sigma=j)}{P(\sigma=k)} = \frac{1-2c\sigma(j)}{1-2c\sigma(k)}\, e^{-(1/4N_0)\,[(1-2c\sigma(j))^2-(1-2c\sigma(k))^2]}\,. \tag{5.20}$$

We may use expression (5.20) to show that even though the (8,6,3) code has a DSV of one greater than the MS43, the extreme values of the DS of the (8,6,3) do not contribute as much to the error probability as the extreme values of the MS43 for a broad range of distortion coefficients and signal-to-noise ratios. Upon substituting $\sigma(j)=5/2$ for MS43 and $\sigma(k)=3$ for the (8,6,3) into (5.20), we find that these two DS values would contribute equally to the error probability if

$$\frac{P(\sigma=6)}{P(\sigma=7)} = \frac{1-5c}{1-6c}\, e^{-9c(c-2/9)/4N_0} \tag{5.21}$$

Expression (5.21) indicates that the comparison of the two codes is sensitive to

both the channel distortion and the SNR. Because we have consistently used c=.1 in our examples we shall perform the comparison for this value. The appropriate probabilities for the two codes are: .000371 for (8,6,3) and .0186 for MS43. The substitution of these values into (5.21) yields

$$\frac{.0186}{.000371} = 1.25e^{.0275/N_0},$$

and solving for N_0, we find that the extreme value of the DS for the (8,6,3) code will contribute more to the error probability than that of the MS43 only when the SNR is greater than 21.3 db.

This analysis demonstrates the effect of the DS probability distribution on the noisy DS channel. Although in the limit of high SNR (or severe distortion), a code with a greater DSV will produce a greater error probability than one with a lesser DSV, for a specific application a small probability of the extreme DS value in the code with larger DSV may be sufficient to reduce the conditional error probability of the extreme value. Furthermore, even if the greater DSV priduces a greater error probability, it may be acceptable for proper system performance. A similar comparison between the two extreme DS values ($\sigma = 1$ and $\sigma = 2$) of the (8,6,3) code reveals that the extreme DS value produces less error probability than its adjacent DS value whenever the SNR is less than 47 db (for c=.1). For the MS43 code, however, this cutoff is only 14.7 db.

We will now apply the ideas discussed in this section to the design of a (7,5,4) code. From Table 6, we see that the minimum number of states for such a CE code is four. The first step in any CE code construction is the construction of the sets of k-eligible words, where k=1,2,3,4 in this case. Although we have demonstrated that the DSV is not in itself an infallible criterion, we will use

it for a first iteration in this procedure. We will want to determine the minimum DSV possible for a (7,5,4) code, which is obviously greater than or equal to four because each state corresponds to a different DS value in the first digit of each codeword. Moreover, if we determine the minimum DS value in the positive reflecting mapping, and design a balanced code, the maximum value (hence the DSV) will be determined. To find the minimum value of the DS, the set of 1-eligible words, A_1, is transformed into an equivalent set of DS sequences of length 7. These sets are defined as

$$D_k = \{(k,\sigma_1,\sigma_2,\ldots,\sigma_7) \mid \sigma_i=\sigma_{i-1}+a_{i-1};\ i=2,\ldots,7;\ \underline{a}\in A_k\}. \tag{5.22}$$

Note that it is not necessary to include the additive constant, C , in the construction because we are interested in the number of DS values, d, and not the actual values of the DS. From the set of 1-eligible words (in which there are 140 words), we must choose $2^7=128$ words such that the smallest value of σ_i ; $i=1,\ldots,7$, is as large as possible. After this operation, we find that the minimum value of the DS in state one is zero. Because state 4 is the complement of state 1, the maximum value is 5; and therefore, d=6. Now that the minimum DSV has been established, we eliminate all code words in A_1 and A_2 for which there is an occurrence of a DS value less than zero in D_1 and D_2, respectively. We have reduced the number of eligible words for the codebook in order to ensure d=6. For the next iteration in the design process, we will choose codewords for state 1 and state 2 in order to minimize the probability of the extreme values of the DS. Because the code is balanced, both extreme values contribute equally to the error probability.

In Chapter 1, we indicated that the probability of a DS value is the average of the conditional DS probabilities over the set of encoder states. The conditional probability for state i is

$$P(\sigma=j|\text{state } i)=(1/n)p_i \sum_{u=1}^{2^m} \theta_u \sum_{q=1}^{n} \phi_{iu}^{(q)}(j), \tag{5.23}$$

where p_i is the stationary probability of the encoder state i, θ_u is the probability of the binary input block, b_u, and

$$\sum_{q=1}^{n} \phi_{iu}^{(q)}(j)$$

is the number of occurrences of the DS value j in the codeword in state i corresponding to input block b_u. In the code synthesis problem we will assume that each binary input block is equiprobable (i.e., $\theta_u=2^{-m}$), and consider the conditional DS probability as

$$(1/n)2^{-m}p_i \; \Phi_i(j)$$

where $\Phi_i(j)$ is the number of occurrences of the DS value j in state i. In the case of the (7,5,4) code that we are constructing, $\Phi_i(j)= \Phi_{r-i+1}(d-j+1)$ and $p_i=p_{4-i+1}$, because the code is balanced. The synthesis problem is the selection of the codewords for state 1 and state 2 such that $p_i \; \Phi_i(l)$ is minimized for i=1,2 and $l \in \{0,5\}$. We will proceed heuristically with this minimization.

First, we note that there will be more occurrences of the zero DS value in state 1 than state 2 because the DS sequence for a codeword in state 1 is one less in every digit than the DS sequence for the same word in state 2 (see (5.22)). It is desirable, therefore, to minimize p_1, which is the first

component of the vector $\underline{p}$ which solves

$$\underline{p}\Pi = \underline{p}$$

where Π is the state transition matrix for the encoder. In order to minimize p_1, we shall choose codewords which minimize the encoder state transition probabilities π_{j1}, j=1,2 which is equivalent to choosing as many words as possible with characteristics equal to one and two in state 1. Upon checking the set of 1-eligible words, we find 44 potential codewords having characteristic equal to one and 30 words having characteristic equal to two. Because we need 128 words to complete the encoding mode for state 1, we still need to choose 54 words, all of which will have characteristic equal to zero or three. We choose these 54 words from the remaining 1-eligible words according to the number of zero (and five) DS values they contain in order to keep the number of occurrences of zeroes and fives as small as possible. We find 33 such words which contain no zeroes, and 14 which contain one zero. We choose the remaining seven words from those which contain two zeroes. We note that in this case, it is impossible to fill the encoding mode such that the DS transition probability, q_{11}, is zero. However, we have chosen the words in a manner which minimizes this probabiity.

In order to fill the encoding mode for state 2, we refer to the set of 2-eligible words and its associated set of DS sequence blocks, D_2. Our selection rule to minimize the encoder state transitions to state 1 (and state 4) dictates the selection of all eligible words having characteristic equal to zero and one, of which there are 95 such words. This leaves 33 words to be chosen from the eligible codewords having characteristic equal to -1 and 2. We

will select these words according to the number of occurrences of extreme DS values. In this case, it is easier to maximize the number of occurrences of the DS values 2 and 3 which are the central DS values. There are 18 remaining eligible words which contain only DS values 2 and 3. We choose the last 15 words from the set of remaining 2-eligible words containing only one occurrence of the DS values 1 and 4.

This completes the design of a (7,5,4) code for the DS channel. We note that the resulting code is state independent decodable. We exhibit the codewords in states 1 and 2 for this code in Table 7. The error performance of this code on the DS channel is slightly better than that of MS43, the difference being of minor consequence. However the (7,5,4) code provides a greater rate increase (m/n=1.4) than the MS43 (m/n=1.33). It is interesting to determinine whether a better error performance is possible for a 7/5 code if we increase the DSV by one (d=7), which is the same as that of the (8,6,3) code. To accomplish this, we must design a (7,5,5) code because the minimum increase in the DSV for a balanced 4 state code is two. The positive reflecting mapping for the (7,5,5) code will be identical to that of the (7,5,4) code. The encoding mode for state 2 is only a slight modification of the corresponding mode in (7,5,4). In the 5 state code, we may use all of the words having characteristics equal to two in state 2 because state 4 is not equivalent to state 1 (as it was for the 4 state code). This means that we will use the 11 words having characteristic equal to two which were not used in the 4 state code. We replace the 11 words which contain the largest number of extreme DS values having characteristic equal to -1 in state 2. The new state in the enocoder becomes state 3, and we choose codewords for this state such that the state transition probabilities π_{33} are maximized; that is, all of the 3-eligible words having characteristic equal to zero are used. The remaining 78 words are chosen to have characteristic equal

Table 7. Codebook for the (7,5,4) code

Codewords for state 1

00+0-	00+-0	000+-	0++--	0+00-	0+0-0	0+-+-	0+-00	0+--+	+0+--
+000-	+00-0	+0-+-	+0-00	+0--+	++0--	++-0-	++--0	+-+0-	+-+-0
+-0+-	+-000	+-0-+	+--+0	+--0+	-++0-	-++-0	-+0+-	-+000	-+0-+
-+-+0	-0++-	-0+00	0-++-	0-+00	0-+-+	000-+	00-+0	00++-	00+00
00+-+	000+0	0000+	00-++	0++0-	0++-0	0+0+-	0+000	0+0-+	0+-+0
0+-0+	0-++0	0-+0+	0-0++	+0+0-	+0+-0	+00+-	+0000	+00-+	+0-+0
+0-0+	+++--	++00-	++0-0	++-+-	++-00	++--+	+-++-	+-+00	+-+-+
+-0+0	+-00+	+--++	-0++0	-0+0+	-00++	-+++-	-++00	-++-+	-+0+0
-+00+	-+-++	0-0+0	00++0	00+0+	000++	0+++-	0++00	0++-+	0+0+0
0+00+	0+-++	0-+++	+0++-	+0+00	+0+-+	+00+0	+000+	+0-++	+++0-
+++-0	++0+-	++000	++0-+	++-+0	++-0+	+-++0	+-+0+	+-0++	-0+++
-+++0	-++0+	-+0++	00+++	0+++0	0++0+	0+0++	+0++0	+0+0+	+00++
	++++-	+++00	+++-+	++0+0	++00+	++-++	+-+++	-++++	

Codewords for state 2

00+0-	00+-0	000+-	0++--	0+00-	0+0-0	0+-+-	0+-00	0+--+	+0+--
+000-	+00-0	+0-+-	+0-00	+0--+	++0--	++-0-	++--0	+-+0-	+-+-0
+-0+-	+-000	+-0-+	+--+0	+--0+	-++0-	-++-0	-+0+-	-+000	-+0-+
-+-+0	-0++-	-0+00	0-++-	0-+00	0-+-+	000-+	00-+0	00++-	00+00
00+-+	000+0	0000+	00-++	0++0-	0++-0	0+0+-	0+000	0+0-+	0+-+0
0+-0+	0-++0	0-+0+	0-0++	+0+0-	+0+-0	+00+-	+0000	+00-+	+0-+0
+0-0+	+++--	++00-	++0-0	++-+-	++-00	++--+	+-++-	+-+00	+-+-+
+-0+0	+-00+	+--++	-0++0	-0+0+	-00++	-+++-	-++00	-++-+	-+0+0
-+00+	-+-++	0-0+0	00++0	00+0+	000++	-+-0+	--++0	0++-+	0+0+0
0+00+	0+-++	0-+++	--+0+	--0++	+0+-+	+00+0	+000+	+0-++	-0+-+
-00+0	-000+	-0-++	0-00+	0--++	++-0+	+-++0	+-+0+	+-0++	00-0+
-+++0	-++0+	-+0++	--+++	00-+-	000-0	0000-	00+--	0-+0-	0+--0
	0+-0-	0+0--	-+00-	-++--	+0-0-	+00--	+-00-	+-+--	

to 1. Because the code is balanced, state 3 (which is the central state in this case) must be a balanced mode which requires choosing the codewords having characteristic equal to 1 in complementary pairs. Prior to actually constructing the codebook, we may perform the comparative analysis developed in this section in order to determine whether we will gain an improvement in error performance by increasing the DSV. Because the distribution of the codewords by characteristic in each state is known, the stationary probabilities of the encoder states may be calculated. An estimate of the probability of the extreme DS values may be made based on the number of occurrences of the 0 DS value in states 1 and 2 of the (7,5,4) code. When this analysis was made, we found that for signal-to-noise ratios greater than 16 db., the extreme value for the 5 state code contributes more error probability than that of the 4 state code (for c=.1). Thus, in this case, we have nothing to gain from the more complex code.

5.4 Coding for DSF Equalization

When the Digital Sum Feedback equalization scheme discussed in Chapter 4 is used, the multimodal nature of the I.I. density is not a factor in the error probability. For channels with a slow response, we have seen that the error probability is only slightly greater than that of the channel with no I.I. The slight increase which does occur is a result of the conditional variances of the I.I. In section 4.4, we demonstrated that the conditional variances are strongly dependent on the response duration of the channel (viz., the variances increase as the duration decreases). We do not expect that the conditional variances will be significantly altered through the code design process. However, we may define a practical coding problem for this system by considering the operation of the DS correction device in the feedback loop.

We have mentioned that a single error in the estimated sequence will produce an error of magnitude c in the feedback loop of the equalizer. We know moreover that this error will be corrected when the channel sequence reaches the terminal DS value. The coding problem for this equalization scheme is thus defined to be the selection of a codebook which minimizes the average time until a terminal DS value is reached by the encoder. The first objective of this code design problem has been mentioned; that is, the construction procedure should minimize the number of DS values, d. Because the errors are effectively independent of the channel sequence statistics, we may assume that an error will occur at each DS value with equal probability. In order to minimize the time to correction, it is clear that we must maximize the probability of the extreme DS values.

We will conclude this section with an illustration of the code design process for a (7,5,4) code. The procedure for designing the codebook is similar to the one illustrated in section 5.3, however, the objective is the opposite. We begin with the formation of the sets of eligible words, A_1 and A_2. Recall that the minimum value of d is 6, and we eliminate all codewords in A_1 and A_2 for which there is an occurrence of a DS value less than zero. In this case, we want to maximize the number of occurrences of the 0 and 5 DS values, therefore we should choose all of the 1-eligible words which have characteristic equal to zero and three in order to maximize the stationary probabilities of the extreme encoder states (1 and 4). Upon checking the set A_1, we find 60 such codewords, which leaves 68 codewords to be chosen for this encoding mode. In the next pass, we choose words containing DS values of 0's and 5's from the remaining 1-eligible words whereby we find 19 more codewords. The final 49 words are selected from the remaining 1-eligible words according to the number of DS values 1 and 4 which occur (viz., those words which maximize the number of

occurrences of these values).

When choosing codewords for state 2, we wish to maximize the number of encoder state transitions to the extreme states, 1 and 4. This indicates that all codewords having characteristics equal to -1 and 2 should be chosen. Unfortunately, we may not blindly fill this mode with these codewords because the final codebook will not be state independent decodable. We have used all except five codewords having characteristic equal to 0 in state 1. Because the code is balanced, all codewords except 5 having characteristic equal to zero will be elements of state 4 (the five exceptions being the complements of the unused words in state 1). The SID sufficient condition (5.9) requires that all codewords which are used in states 1 <u>and</u> 4 must also be used in states 2 and 3. Therefore, we must include the 45 eligible codewords in state 2 having characteristic equal to zero except for the 5 codewords which do not appear in both states 1 and 4.

Next we may choose the 30 codewords having characteristic equal to 2 because there is no conflict with the SID condition. This leaves 58 words to be chosen in order to complete the encoding mode for state 1. It would be desirable to use all of the codewords having characteristic equal to -1, however, this would conflict with the SID requirement. To satisfy the SID condition, we begin by choosing those words having characteristic equal to -1 whose complements were not used in state 1, of which there are seven codewords. For the remaining 51 words, we must choose chodewords such that every codeword having characteristic equal to -1 appears with its complement. Therefore we may choose 25 complementary pairs from the remaining 2-eligible words. The last codeword must be chosen from the remaining eligible words having characteristic equal to one. The final 51 codewords are chosen such that the number of

occurrences of the extreme DS values is maximum.

The code obtained from this design procedure has an error performance for the DSF equalized channel which is slightly worse (approximately .3 db) than that of the MS43 (see Fig. 20 in Chapter 4). The probability of the extreme DS values is .0374 as opposed to .0173 for the (7,5,4) code designed in section 5.3. We see that for these parameters, the range of the extreme DS probabilities is not great. This is not surprising because the amount of choice available for the positive reflecting mapping is not great (128 words out of 136 eligible when d=6). Furthermore, the SID requirement is a further constraint when we attempt to maximize the extreme DS probabilities.

CHAPTER 6

DECODING AND ERROR CONTROL AT THE TERMINAL

6.1 Decoding at the Terminal

We concluded Chapter 2 with a short discussion the model for the destination terminal of the repeatered line (see Fig. 8). We may recall that the terminal is identical to a repeater with the exception of a decoder which converts the received (estimated) channel sequence into an estimate of the original binary information sequence. In this section, we shall discuss the structure of this decoder, and mention some aspects of the error performance of the decoder relative to the error performance of the channel symbol sequence.

Because the encoder processes the original binary information sequence in blocks of m bits and emits blocks of n symbols, the decoder must process the estimated channel symbol sequence in blocks of n symbols. However, the decision device which produces the estimate of the channel sequence is a symbol-by-symbol detector, and thus does not produce a sequence which may be directly decoded. It is necessary, therefore, for the decoder to preprocess the estimated symbol sequence in order to provide a sequence of symbol blocks having length n. This preprocessing is called <u>word synchronization</u> or <u>framing</u>. When the encoder is counter-encodable, the channel sequence contains sufficient redundancy for the word synchronization to be accomplished without additional information (e.g., extra symbols specifically for synchronization). A general method for word synchronization, reported by Preparata and Bellato [11], is the construction of a finite state machine (called the <u>monitor</u>) which contains one state for each subset of states in the encoder, plus an alarm state. When the estimated channel sequence is produced by the detector, it passes through a framing

circuit which divides the symbol sequence into blocks of n-tuples. This block (vector) sequence is then input to the monitor. The monitor is initially in a state which corresponds to the complete set of encoder states (this represents its ignorance of the actual encoder state). As each n-tuple enters the monitor, its characteristic (real sum of the symbols) is calculated and the monitor changes state according to the result. Eventually, the monitor reaches a state corresponding to a single encoder state. From this time on, it mimics the state transitions of the encoder (i.e., there is a sub-automaton imbedded in the monitor which is isomorphic to the encoder). The PST code (see Table 1) provides a simple example. The state diagram of its monitor is illustrated in Fig. 25. The sub-automaton is shown in the box and the alarm state is denoted by C^*. The numbers associated with the state transitions are the characteristics of the received codewords.

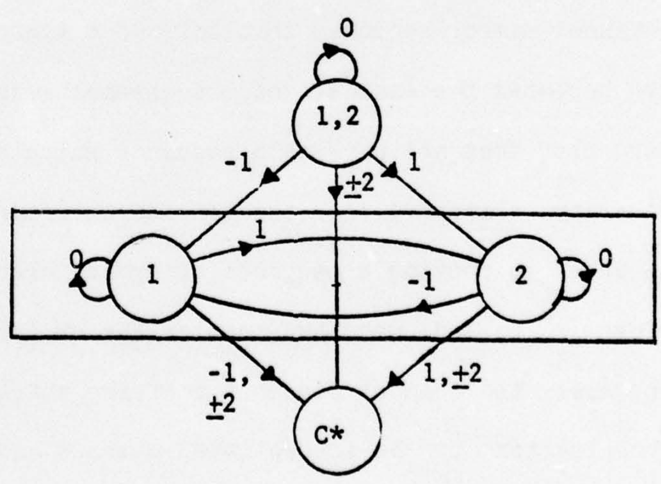


Figure 25. Monitor for PST

Suppose that the monitor is in its initial state, {1,2}, and it receives the block '0+'. It calculates the characteristic, +1, and changes to the state corresponding to the encoder state 2, {2}. The monitor in this case has become

synchronized with the encoder because the only way the encoder could have produced the sequence 0+ is if it had been in state 1; after producing the codeword '0+', its next state would have been state 2. In the absence of transmission errors, the monitor would remain synchronized with the PST encoder.

If there are transmission errors, or if for any reason the word synchronization is lost (i.e., the n-tuples entering to monitor do not correspond to codewords, but rather the concatenation of symbols from two adjacent codewords), the characteristics of these n-tuples will eventually violate the encoder state transition rule. When this occurs, the monitor enters the alarm state. It then signals a device which we call the framing strategy device which must decide if the violation occurred as a result of a transmission error or a misframed condition. The monitor then enters its initial state and attempts to become synchronized with the encoder again. The framing strategy device must also have the capability to adjust the framing circuit when it decides that a misframed condition prevails. Fig. 26 illustrates a block model of the word synchronization system.

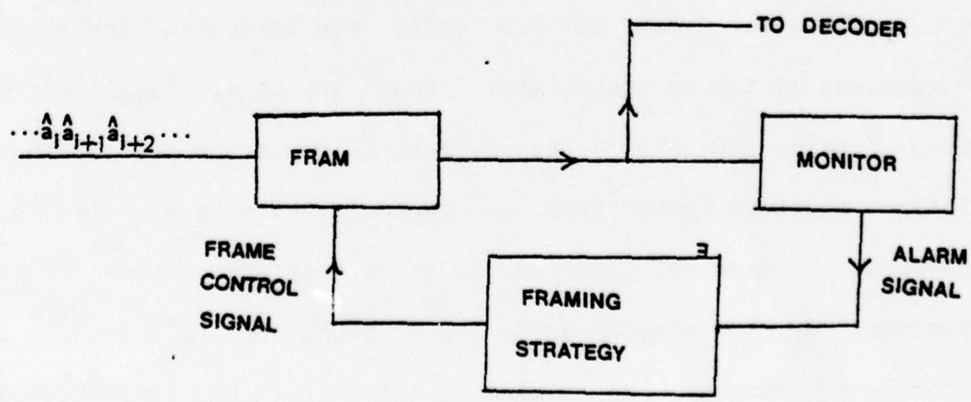


Figure 26. Word synchronization system

The strategy employed for determining the word synchronization is based on the statistics of the encoder. For codewords of length n, there is one correct frame position (phase) and n-1 incorrect phases. The statistics of the state transition violations for each of the incorrect phases are used to determine the expected frequency of occurrence of these violations. The framing strategy box in Fig. 26 identifies the frame condition based on these statistics and transmits the appropriate control signal to the framing circuit in order to obtain word synchronization. A specific framing strategy for the MS43 code, which is based on a simplified encoder state monitor, has been reported in [28].

When the symbol sequence has been properly framed, the blocks of n-tuples may be converted into the corresponding blocks of binary m-tuples by the decoder when no transmission errors have occurred. Because the CE codes are non-linear in general, a table lookup scheme is necessary to accomplish this conversion. This is not a difficult task because the codes under consideration are short ($n \leq 7$), for which current LSI technology is sufficient. The n-tuples are converted to addresses in a read-only-memory which contain the binary m-tuples.

In Chapter 5, we introduced the property of state independent decodability for a CE code. For an SID code, the inverse of the encoding function is independent of the encoder state; that is, $b_u = f^{-1}(\underline{a}_{iu})$ where $\underline{a}_{iu} = f(s_i, b_u)$. When decoding an SID code, we must allocate one address in the ROM for each distinct codeword (note that in general there are more than 2^m distinct codewords). However, if the code is not SID, then 2^m addresses must be allocated for each encoder state for a total of $r2^m$ 2n-bit addresses. The memory requirements are always greater for the state dependent decoding in practical cases. Moreover, when the decoding is state dependent, the state identification monitor must also be incorporated into the decoding operation.

Because the word synchronizer estimates the state of the encoder (after the identification transient), the estimate may be provided to the decoder for use in state dependent decoding (i.e., the estimate tells the decoder which decoding table to use). We will see later that this dependence on the state identification monitor can have detrimental effects on the error probability, but can be used to advantage in an error detection scheme.

In the previous chapters, we have been concerned with the channel symbol errors in the repeater decision device when the channel sequence is disturbed by Gaussian noise and I.I. is present. It is clear that there is a direct relationship between the channel symbol errors and the bit errors at the output of the decoder. Because the original information sequence was encoded into blocks of channel symbols, a single error in the channel sequence may cause as many as m bit errors in the decoded information sequence. Thus, we may consider the bit error probability, denoted $P_b(E)$, to be $mP(E)$. This error propagation as a result of the block encoding is not serious as long as the block length of the code is short (an order of magnitude in error probability corresponds to approximately 1 db. of reduction in the SNR for applications we have discussed). This approximation of $P_b(E)$ is valid when we assume that the decoder operates on the properly framed block of channel symbols. When the channel sequence is improperly framed, the probability of a bit error is very high (close to one). Fortunately, there is enough redundancy in the channel sequence to detect many error patterns. We discuss the various error detection schemes briefly in the following section.

6.2 Error Detection in the Terminal

We have mentioned error monitoring in the repeaters in Chapter 4 when the DSF equalization scheme was discussed. In addition to this, we may detect errors in the decoder with either the state identification monitor or the table lookup procedure (or both).

Error detection in the state identification monitor begins after the monitor becomes synchronized with the encoder. The error patterns which may be identified are a function of the state of the monitor/encoder. The condition for detection is that the error pattern must change the characteristic of the n-tuple into a value which does not occur in the encoder when the encoder is in its present state. For example, in the simplest case of the PST code, if the encoder and monitor are in state 1, the error pattern must produce a pair of symbols which has characteristic equal to -2,-1, or 2 in order to be detected. The effectiveness of this error detection scheme is related to the number of state transitions which may occur from each encoder state. If an encoder has a minimal number of states and the codewords are chosen in order to minimize the number of states to which it may change, the error detection will be most effective. We would expect, therefore, that an encoder having only one state (in which all words have characteristic equal to zero, of course), would be the most effective because all errors which change the characteristic of the received n-tuple from zero are detectable.

A disadvantage of this error detection scheme may be recognized from the action of the state identification monitor after it detects a violation. We may recall that the monitor enters the alarm state, and signals the framing strategy device. It then enters its initial state and begins the state identification operation. The error detection capability of the monitor is reduced until it

becomes synchronized with the encoder. It must be noted, however, that it still may provide error detection during the state identification phase of its operation. A detailed quantitative analysis of the probabilities involved has not been reported.

Error detection schemes implemented in the table lookup procedure are possible for CE codes because in general the complete set of n-tuples do not appear in the codebook. The error detection is accomplished by allocating one address in the ROM for each of the 3^n n-tuples. The contents of those addresses which can not be generated by the encoder outputs contain the error message. The effectiveness of this scheme is dependent on whether state dependent or state independent decoding is performed. When the decoding is state independent, an error is detected whenever the received n-tuple does not appear anywhere in the codebook. The number of received n-tuples which will generate the error message is 3^n-N, where N is the number of distinct words in the codebook and is larger than 2^m in general. For example, the MS43 codebook contains every 3-tuple except '000', and therefore, the SID error detection scheme would not be very effective. If the decoding is state dependent, then there are 3^n-2^m n-tuples which would indicate an error, providing greater error detection capability. Of course, more memory is required in the state dependent decoding table. For a given code, the effectiveness of error detection in the monitor versus that in the table lookup would need to be analyzed to determine if the greater complexity in the table lookup is warranted. We also mention that a combination of error detection in the monitor and in the table lookup procedure may be more efficient. The memory space necessary in the table would be reduced to the number of n-tuples which have the same characteristic as the codewords in each encoder state.

We must emphasize that even if state dependent decoding is used, it is most desirable that the code be *SID*. If a code is not SID, correct decoding is dependent of the operation of the monitor. Whenever an error is detected by the monitor, the decoding process is interrupted until the monitor regains synchronization with the encoder. Thus an error which is confined to a single n-tuple will disable the decoder for an indefinite length of time. We see that although state dependent decoding may be desirable from the standpoint of error detection, it is detrimental to the decoding process if the code is not SID.

In this section we have presented a general discussion of error detection methods for CE codes. In a given situation, the tradeoffs involved would need to be considered, and a more detailed quantitative analysis performed in order to determine the best method. In the next section, we will take these error control considerations one step further, and consider single error correcting CE codes.

6.3 Illustration of Single Error Correcting CE Codes

The correction of errors in the encoder/decoder pair with CE codes is possible without a great reduction in the transmission rate of the channel symbols. In this section, we will apply a variation of a known construction for nonlinear single error correcting codes to demonstrate the existence of CE codes which correct single adjacent errors.

We define a <u>single adjacent error</u> in a codeword containing n ternary symbols as the vector $\underline{e}=(e_1, e_2, \ldots, e_n)$ where $e_j=\pm 1$ for some $j=1,\ldots,n$, and $e_i=0$ for $i\neq j$. When a single adjacent error occurs in the j^{th} symbol of a codeword $\underline{a}$, the received symbol which has been altered must still equal -1,0, or 1. The

definition of the single error is identical to the definition of a single error given in section 4.2 with the additional constraint that only one such error occurs within a block of n symbols corresponding to a codeword. Note that this definition is weaker than the definition of symmetric errors usually encountered in the coding literature. In Fig. 27, we illustrate the bipartite graphs which characterize the ternary symmetric channel, and the adjacent error channel.

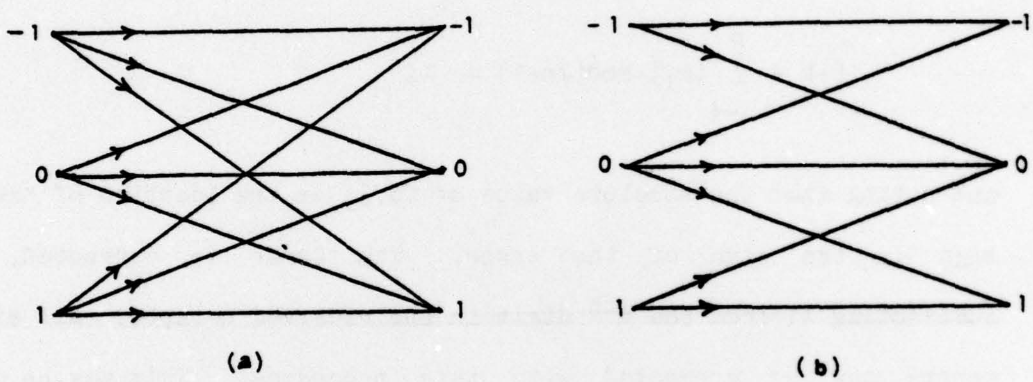


Figure 27. (a) Ternary symmetric channel, (b) Adjacent error channel

The errors we are discussing are not symmetric because each ternary symbol is not subject to all possible errors; the zero symbol may be altered by ± 1, however, the nonzero symbols, ± 1, may only be altered by ∓ 1, respectively. First, we describe a method of constructing nonlinear ternary codes which are capable of correcting single adjacent errors[29]. It is similar to a construction given in [30] for codes which are capable of correcting asymmetric errors (a weaker property than adjacent errors). We define the code as the set of ternary n-tuples, $\underline{a}=(a_1, a_2, \ldots, a_n)$, such that

$$\sum_{i=1}^{n} ia_i \equiv b \mod(2n+1), \tag{6.1}$$

where b is a constant with absolute value less than n, and $a_i \in \{-1,0,1\}$. If we assume that a single error occurs in the i^{th} digit when the codeword $\underline{a}$ is transmitted, the received n-tuple is

$$\underline{r} = (a_1,\ldots,a_{i-1},a_i\pm 1\ ,a_{i+1},\ldots,a_n). \tag{6.2}$$

We may correct this error by computing the quantity ("syndrome")

$$[-b + \sum_{i=1}^{n} ir_i] \bmod(2n+1) = \pm i \tag{6.3}$$

and noting that the absolute value of (6.3) is the location of the error and the sign is the sign of the error. The error is corrected, therefore, by subtracting ± 1 from the i^{th} digit in the received n-tuple. All single adjacent errors may be corrected with this procedure. This may be demonstrated by considering the decoding operation on two received n-tuples; one with an error in position i, and the other in position j. The two syndromes, (6.3), will be identical iff $i \equiv j \bmod(2n+1)$ in the case where the errors have the same sign, and iff $|i|+|j| \equiv 0 \bmod(2n+1)$ in the case where the errors have opposite sign. The former case is true only when i=j because the magnitudes of both quantities are less than n, and the latter case is never true because the sum of the magnitudes is always less than 2n+1.

It is interesting to note that a lower bound on the number of codewords for a code constructed by this procedure is the same as the Hamming upper bound for ternary single symmetric error correcting codes. If

$$\sum_{i=1}^{n} ia_i \bmod(2n+1)$$

is computed for each of the 3^n ternary n-tuples, and the n-tuples grouped together according to the result, the pigeon-hole principle assures us that there will be at least $3^n/(2n+1)$ n-tuples which have the same value of b, for some b.

We will use this construction to show the existence of a few short CE codes which are capable of correcting a single adjacent error while not requiring a great penalty in the symbol transmission rate, m/n. A computer program has calculated the number of codewords which may be used in such a code for blocklengths up to seven. The set of possible n-tuples having non-negative characteristics was first partitioned into (n+1) sets, with each set corresponding to a distinct characteristic 1,...,n, and a set for the words having characteristics equal to zero. The number of words having negative characteristics which have a value b in (6.1) is equal to the number of words with the corresponding positive characteristics having the value -b.

Recall from Chapter 5 that we have constrained the design of CE codes to the class of balanced and state independent decodable codes. A necessary and sufficient condition for state independent decoding of the single error correcting codes is

$$\sum_{i=1}^{n} ia_i \bmod(2n+1) = \sum_{i=1}^{n} i\hat{a}_i \bmod(2n+1) \tag{6.4}$$

for two codewords, $\underline{a}$ and $\hat{\underline{a}}$, being elements of distinct states. Moreover, we know that in a balanced code, if codeword $\underline{a}$ appears, then its complement, $\bar{\underline{a}}$, also appears somewhere in the codebook. In order to satisfy expression (6.4), a balanced code must contain only codewords for which expression (6.1) equals zero (b=0). Table 8 contains the number of codewords for each characteristic which

have this property for blocklengths equal to 5, 6, and 7.

Table 8. Number of available codewords for SEC balanced codes

LENGTH	CHARACTERISTIC							
	0	1	2	3	4	5	6	7
5	6	4	1	2	1	0	-	-
6	14	9	4	5	1	0	0	-
7	32	23	13	11	8	1	0	0

From Table 8, we find that the best length-5 code is a (3,5,2) code for which there are two extra words available for the positive reflecting mapping. This code is guaranteed to be SID because all 2-state codes satisfy the sufficient condition (5.9). The highest rate code for $n=6$ is a (5,6,4) code for which there are just 32 codewords available for the positive reflecting mapping. An SID code may be constructed by using all codewords having characteristics equal to zero and one in the mapping for state 2. Finally, for $n=7$, the highest rate code is a (6,7,3) code for which there are 4 extra words available for the positive reflecting mapping. We can also ensure the SID property in this case by using all codewords having characteristic equal to zero in the mapping for state 2.

CHAPTER 7

SUMMARY AND CONCLUSIONS

In this thesis, we have investigated the relationship between the structure of counter-encodable codes and the symbol error probability for a fixed threshold detector on a linear dispersive additive white Gaussian noise channel. We have shown that the encoder state process may be utilized to summarize the past sequence of channel codewords for the purpose of determinng the probabilty distribution of the I.I. We then specialized this idea to the DS process associated with the channel symbol process. Because the DS assumes values in a finite set, we were able to relax the usual restriction of truncating the channels' time response in order to calculate the error probability via the Gram-Charlier series. We then found a set of functional equations for the conditional characteristic functions of the I.I. for a first order error channel. This allowed us to calculate the moments of the I.I. by recursively solving systems of linear equations, whereby the error probability was calculated by utilizing the conditional moments. The effects of the distortion coefficient, c, and the rate of decay, λ , on the error probability for the first order error channel were illustrated for several known CE codes.. We found that different codes exhibit significantly different error performance. The largest difference in performance occurred for channels with a slow decay rate. We also illustrated that the effect of this decay rate on the performance of a single code is significant with the worst error performance occurring for the slowest rate of decay ($\lambda \simeq 1$).

These results led us to investigate a simple channel model, the DS channel, which was found to produce accurate results for channels with a slow response. The major result arising from the DS channel model is the multimodal probability density function of the I.I. We found one mode for each value of the DS with each mode being equal to the DS value scaled by the channel distortion coefficient, c. Furthermore, we found that when $\lambda < 1$, the assumption that each mode is Gaussian with variance equal to the conditional I.I. variance produced extremely accurate error probability results.

Another result arising from the DS channel model is the design of a simple decision feedback equalization scheme which keeps track of the DS of the dectected channel sequence and compensates for the strong bias in the I.I. We found that the deleterious effects of the I.I. on the error probability were effectively reduced with this scheme. This scheme is effective because it is based directly on the statistics of the I.I. In addition, we discussed a method of controlling the error propagation which is inherent to all decision feedback schemes.

The relatively simple expression for the error probability on the DS channel provides a convenient method of comparing the error performance of different codes. We found that the number of DS values, the DS transition probabilities, and the steady state DS probabilities should be considered before judging code performance. This is a more accurate technique than considering only the number of DS values which is current practice. All of these considerations led to a procedure for constructing CE codes which minimize the error probability on the DS channel. We have shown an example of this construction for a (7,5,4) CE code which exhibits better error performance than the known MS43, yet also provides a greater rate increase. We also illustrated

a coding procedure for the DSF equalization scheme. The objective for these constructions is opposite that for the fixed threshold detector case; however, the construction procedure is similar.

In the last chapter, we discussed some problems associated with decoding CE codes. In particular, a general structure for word framing was presented. We then discussed techniques for error detection at the terminal. Finally, we demonstrated the existence of single error correcting CE codes via a known construction of nonlinear ternary error correcting codes.

REFERENCES

1. H. Nyquist, "Certain Topics in Telegraph Transmission Theory," Trans. AIEE, vol. 47, April, 1928, pp. 617-644.

2. R.W. Lucky, J. Salz, and E.J. Weldon, Principles of Data Communication, New York, McGraw-Hill, 1968.

3. W. R. Bennett, and, J. R. Davey, Data Transmission, New York, McGraw-Hill, 1965.

4. L. E. Franks, editor, Data Communication: Fundamentals of Baseband Transmission, Stroudsberg, Pa., Hutchinson , and Ross, Inc., 1974.

5. R. W. Lucky, "A Survey of the Communication Theory Literature: 1968-1973," IEEE Trans. Info. Thy., vol. IT-19, no. 5, November, 1973, pp. 725-739.

6. H. Kobayashi, "A Survey of Coding Schemes for Transmission or Recording of Digital Data," IEEE Trans. Commun. Tech., vol. COM-19, December, 1971, pp. 1087-1100.

7. A. Croisier, "Introduction to Pseudoternary Transmission Codes," IBM J. Res. Develop., vol. 14, July, 1970, pp. 354-367.

8. F. S. Hill, Jr., and M.A. Blanco, "Random Geometric Series and Intersymbol Interference," IEEE Trans. Info. Thy., vol. IT-19, May, 1973, pp. 326-335.

9. P. J. McLane, "Upper and Lower Bounds for Error Probability in Pulse Communication Systems," Proc. 13th Allerton Conf. on Circuit and System Theory, October, 1975, pp. 782-792.

10. G. L. Cariolaro, and G. P. Tronca, "Spectra of Block Coded Digital Signals," IEEE Trans. Comm., vol. COM-22, no. 10, October, 1974, pp. 1535-1563.

11. F. P. Preparata, and L. Bellato, "Error Detection and Synchronization with Pseudoternary Codes for Data Transmission," Alta Frequenza, vol. XLII, no. 6, 1973, pp. 280-285.

12. J. M. Sipress, "A New Class of Selected Ternary Pulse Transmission Plans for Digital Transmission Lines," IEEE Trans. Commun. Tech., vol. COM-13, no. 3, September, 1965, pp. 366-372.

13. E. Y. Ho, and Y. S. Yeh, "A New Approach for Evaluating the Error Probability in the Presence of Intersymbol Interference and Additive Gaussian Noise," B.S.T.J., vol. 49, no. 9, November, 1970, pp. 2249-2265.

14. O. Shimbo, and M. I. Celebiler, "The Probability of Error Due to Intersymbol Interference and Gaussian Noise in Digital Communication Systems," IEEE Trans. Commun. Tech., vol. COM-19, no. 2, April, 1971, pp. 113-119.

15. H. Cramer, Mathematical Methods of Statistics, Princeton, N. J., Princeton Univ. Press, 1966.

16. E. Y. Ho, and Y. S. Yeh, "Error Probability of a Multilevel Digital System with Intersymbol Interference and Gaussian Noise," B.S.T.J., vol. 50, no. 3, March, 1971, pp. 1017-1023.

17. G. L. Cariolaro, and S. G. Pupolin, "Moments of Correlated Digital Signals for Error Probability Evaluation," IEEE Trans. Info. Thy., vol. IT-21, no. 5, September, 1965, pp. 558-568.

18. A. Papoulis, Probability, Random Variables, and Stochastic Processes, McGraw-Hill, 1965, p. 157.

19. M. R. Aaron, "PCM Transmission in the Exchange Plant," B.S.T.J., vol. 41, no. 1, January, 1962, pp. 99-141.

20. P. A. Franaszek, "Sequence-State Coding for Digital Transmission," B.S.T.J., vol. 47, no. 1, January, 1968, pp. 143-157.

21. V. DiEuliis, and F.P. Preparata, "Spectrum Shaping with Alphabetic Codes with Finite Autocorrelation Sequence," IEEE Trans. Comm., vol. COM-26, no. 4, April, 1978, pp. 474-478.

22. A. Croisier, "Compatible High-Density Bipolar Codes: An Unrestricted Transmission Plan for PCM Carriers," IEEE Trans. Commun. Tech., vol. COM-18, no. 3, June, 1970, pp. 265-268.

23. P. Bylanski, and D.G.W. Ingram, Digital Transmission Systems, IEE, 1976, pp. 265-267.

24. V. I. Johannes, "Comments on 'Compatible High-Density Bipolar Codes: An Unrestricted Transmission Plan for PCM Carriers,'" IEEE Trans. Comm., vol. COM-20, no. 1, February, 1972, pp. 78-79.

25. V. I. Johannes, A. G. Kaim, and T. Walzman, "Bipolar Pulse Transmission with Zero Extraction," IEEE Trans. Commun. Tech., vol. COM-17, no. 2, April, 1969, pp. 303-310.

26. A. Huzii, and H. Sugiyama, "Intersymbol Interference of Markov Pulse Trains," Elect. and Comm. in Japan, vol. 53-A, no. 3, 1970, pp. 21-30.

27. A. Jessop, and D. B. Waters, "4B-3T, An Efficient Code for PCM Coaxial Line," XVIII Inter. Meeting on Elect., Rome, 1970.

28. O. Brugia, M. Decina, and U. DeJulio, "Character Alignment for High-Capacity PCM Systems Using MS43 Line Code," IEEE Trans. Comm., vol. COM-23, no. 8, August, 1975, pp. 803-812.

29. R. J. McEliece, private communication, 1978.

30. R. R. Varshamov, "A Class of Codes for Asymmetric Channels and a Problem from the Additive Theory of Numbers." IEEE Trans. Info. Thy., vol. IT-19, no. 1, January, 1973, pp. 92-95.

31. J. M. Wozencraft, and I. M. Jacobs, Principles of Communication Engineering, New York, John Wiley and Sons, Inc., 1967, p. 83.

32. S. Karlin, A First Course in Stochastic Processes, New York, Academic Press, Inc., 1966.

VITA

Val Anthony DiEuliis was born in Downingtown, Pennsylvania on September 7, 1950. He received the B.S. degree in Electrical Engineering from the University of Notre Dame in 1972. From 1972-74, he served in the U.S. Army as a project engineer with the Army Security Agency. Since 1974, he has been a Graduate Research Assistant with the Coordinated Science Laboratory at the University of Illinois at Urbana-Champaign, where he received the M.S. degree in Electrical Engineering in 1976.

Mr. DiEuliis is a member of the Institute of Electrical and Electronics Engineers.